



ROBOT EDUCATIU PER UN PBL EN ENGINYERIA INFORMÀTICA.

Memòria del projecte Fi de Carrera
d'Enginyeria en Informàtica
realitzat per

Daniel Lozano Miguel

i dirigit per

Joan Oliver Malagelada

Bellaterra, 18 de Setembre de 2008

El sotasignat, Joan Oliver Malagelada

Professor/a de l'Escola Tècnica Superior d'Enginyeria de la UAB,

CERTIFICA:

Que el treball a què correspon aquesta memòria ha estat realitzat sota la seva
direcció per en

I per tal que consti firma la present.

Signat:

Bellaterra, 18 de Setembre de 2008

Agraïments

Vull agrair a la meva família tot el suport i comprensió que m'han donat en la realització d'aquest projecte i durant tota la carrera, i agrair també l'esforç que han fet per poder-me donar una bona educació.

A, Joan Oliver per guiar-me al llarg d'aquest projecte, fent que les coses es veiessin més fàcils i divertides.

Per últim, vull agrair als meus company i amics que m'han aguantat i ajudat al llarg d'aquests anys de carrera, fent que el temps transcorris sense adonar-nos i fent d'aquest temps la millor experiència de la meva vida.

ÍNDIX	4
1 INTRODUCCIÓ	7
1.1 Objectius	7
1.2 Motivació	8
2 ANÀLISIS DE REQUERIMENTS	9
2.1 Requeriments funcionals.....	9
2.2 Requeriments físics.....	11
2.3 Costos.....	11
2.4 Planificació	12
3 MODEL SCRIBBLER.....	14
3.1 Hardware.....	14
3.1.1 Mides y conexions	15
3.1.2 Sensors.....	15
3.1.3 Placa Controladora	15
3.1.4 Alimentació	16
3.2 Software	16
3.2.1 Editor gràfic scribbler.....	17
3.2.2 Editor de text scribbler	17
3.3 Problemes / Innconvenients	18
4 ESTUDI D'ALTERNATIVES	20
4.1 Utilització de “Hacker Port”	20
4.2 Alternativa Boebot.....	21
4.3 Cross compiler – Assemblador→PBasic	21
4.4 Recrear Scribbler amb un Microcontrolador	22
4.5 Elecció de l’alternativa	22
5 ESTRUCTURA I DISSENY DEL ROBOT.....	23
5.1 Model Físic	23
5.1.1 Estructura.....	23
5.1.2 Locomoció.....	24
5.1.3 Alimentació	24
5.1.4 Control.....	24

5.1.5	Sensors.....	25
5.2	Unitat de control	26
5.2.1	AVR (Atmel)	26
5.2.1.1	Arquitectura.....	27
5.2.1.2	Eines de desenvolupament	28
5.2.2	PIC (Microchip Technology)	28
5.2.2.1	Arquitectura.....	29
5.2.2.2	Eines de desenvolupament	30
5.2.3	FREESCALE (MOTOROLA)	30
5.2.3.1	Arquitectura.....	31
5.2.3.2	Eines de desenvolupament	32
5.2.4	Exemples	32
5.2.5	Conclusions	33
5.2.6	Elecció	33
5.2.7	ET-Base AVR ATmega128.....	35
5.3	Unitat de sensors	36
5.3.1	Infraroigs	36
5.3.1.1	Detector obstacles	37
5.3.1.2	Seguidor de línies	37
5.3.2	Fotoresistència (LDR)	38
5.3.3	Compàs Magnètic	39
5.3.4	Servomotors.....	40
5.4	Software de programació	43
5.4.1	AVR Studio 4 - WinAVR.....	43
5.4.2	Drivers de control dels sensors.....	44
6	RESULTATS i CONCLUSIONS.....	45
6.1	Millores i Possibles Ampliacions.....	45
7	BIBLIOGRAFIA	47
8	APÈNDIX I.....	49
8.1	Especificacions tècniques	49
8.1.1	Sensor Infraroig - SHArp GP2Y0D805Z0F.....	49
8.1.2	Detector - InfraRoig PNA4601M.....	52
8.1.2.1	Emissor infraroig – QEC112.....	55

8.1.3	Fotoresistència	57
8.1.4	Compàs magnètic – CMPS03.....	58
8.1.5	Servomotors.....	63
8.1.6	Placa ETT-Base AVR ATMega128	64
8.2	Llista de costos.....	66
9	APÈNDIX II	69
9.1	Manual D'Usuari	69
9.1.1	Instal·lació i configuració del Software.....	69
9.1.2	AVR Studio	70
9.1.3	Programació.....	72
9.1.4	Drivers	73

INTRODUCCIÓ

Les noves tecnologies estan cada cop més presents en la nostra societat, utilitzant-se cada cop més i en més aspectes de la vida quotidiana, el que facilita enormement moltes tasques diàries de la vida de les persones. Un d'aquests aspectes d'aplicació de la robòtica és en el món de l'educació.

La robòtica aplicada a l'educació, el que podríem anomenar *robòtica educativa*, és cada dia més utilitzada com a eina d'aprenentatge i cerca de coneixements pràctics, tant per continguts tecnològics, com altres matèries relacionades. Cada cop més centres docents incorporen la robòtica, ja sigui en el seu pla docent o com a activitat extraescolar, on els alumnes comencen a introduir-se en el món de la robòtica.

Objectius

En un principi, l'objectiu d'aquest projecte era l'aprofundiment en diferents conceptes del món de la robòtica, així com l'estudi de les seves possibles aplicacions.

A mesura que va anar avançant el treball del projecte, es van considerar diferents aplicacions, tot mirant d'incorporar en tot moment la idea de la robòtica mòbil al món de la investigació i l'educació, arribant finalment a l'aplicació que hem dut a terme. El principal objectiu d'aquest projecte és l'estudi de la viabilitat, el disseny, i construcció d'una plataforma robòtica educativa per l'aprenentatge i pràctica de diferents conceptes informàtics.

La finalitat del projecte, és la construcció i incorporació del robot dissenyat a l'àmbit educatiu universitari, fent ús del robot com a complement educatiu de diverses assignatures de primer curs d'enginyeria informàtica. La plataforma robòtica ha de ser capaç de permetre l'aprenentatge tant de conceptes de hardware com de software, podent d'aquesta manera aprendre conceptes d'estructura de computadors, conceptes de programació i d'estructura de dades,....., per poder ser utilitzat com complement de diferents assignatures.

Des de que som petits veiem els robots com se'ns mostren a la televisió, maquines molt complexes que tenen capacitat de moviment i un comportament molt intel·ligent semblant al de l'ésser humà. Res més lluny de la realitat. Dotar a un robot de petites característiques per fer el seu comportament autònom, és un treball molt costós i complicat.

Dins del món de la informàtica i l'enginyeria, sempre m'he sentit atret per tot el relacionat amb el hardware i la robòtica, tot i no tenir gaires coneixements dins aquest món, i amb ganes d'introduir-me i aprendre més coses relacionades. En l'elecció d'aquest projecte, se'm presentava una gran oportunitat per introduir-me i veure més proper allò que fins ara només hi havia esta curiositat.

Un cop escollit l'àmbit o temàtica del projecte, vaig pensar que podria ser molt útil l'aplicació dels coneixements adquirits per mirar de dissenyar una eina d'aprenentatge pràctic per als nous alumnes, que fes més interactiu i amè l'aprenentatge de nous conceptes informàtics.

Abans de començar amb el disseny del robot hem de tenir en compte una sèrie de consideracions i requisits previs que determinaran en certa manera el disseny final del robot.

Requeriments funcionals

- Robot autònom mòbil. Això vol dir que necessitem que el robot es pugui comportar d'una manera autònoma un cop programat un comportament, és a dir, que no calgui estar donant ordres a cada moment, sinó que tingui un cert grau d'interacció amb l'entorn i prengui decisions senzilles basades en cada situació. Aquesta interacció comporta dos factors importants:
 - Capacitat de moviment. A de ser capaç de moure's de manera senzilla per un entorn controlat.
 - Capacitat de percepció. Hem de tenir alguns dispositius per recollir informació del medi en el qual ens movem i que ens permetin interactuar amb ell.
- Aplicació en el camp de l'educació. Com ja s'ha mencionat en punts anteriors, aquesta aplicació està destinada al camp de l'aprenentatge. Aquesta característica implica la necessitat d'algunes funcionalitats de cara a les assignatures que poden fer ús del robot per explicar els conceptes presents al seu temari. A continuació es comenten algunes de les funcionalitats que ha de tenir el robot, segons les assignatures relacionades:
 - Fonaments de computadors.

- Arquitectura del computador. Es demana poder estudiar el funcionament intern del computador. Per tant, es vol una arquitectura del computador el més entenedora possible.
- Programació a baix nivell. A de ser possible la programació del microcontrolador en llenguatge ensamblador, per poder observar el comportament del computador a baix nivell.
- Estructura de dades.
 - Ha de ser possible la programació d'alt nivell, ja sigui C o bé python, per tal d'estudiar i aprofundir els coneixements de l'estructura de les dades.
- Algorismes i programació.
 - Ha de ser possible la programació d'alt nivell per tal d'estudiar i aprofundir els coneixements de programació. S'incorporaran diversos sensors per a què l'experiència de programació sigui el més enriquidora possible.
- Altres assignatures.
 - Hem parlat de les principals assignatures relacionades i les que més es volia tractar, però també es poden incloure altres assignatures d'informàtica, com poden ser:
 - Electrònica. Estudi del circuit i de les diferent connexions de components, ed les propietats d'aquests i del seu comportament.
 - Sistemes Operatius. Amb la incorporació d'un S.O en temps real com pot ser el FreeRTOS, es poden estudiar els conceptes de processos, semàfors, planificació de tasques,....

Requeriments físics

- Dimensions adequades. El Robot està destinat a la manipulació per part d'alumnes amb poca experiència, per tant ha de tenir unes dimensions més o menys reduïdes. Això facilitarà el seu estudi i manipulació, millorant d'aquesta manera l'experiència educativa.
- Ja hem dit abans que ha de tenir capacitat de moviment, per tant la disposició de les rodes i motors ha de permetre el seu moviment i donar la capacitat de fer girs i canviar de direcció.
- El robot està destinat a un PBL¹ en informàtica, per tan els usuaris no tindran grans coneixements sobre el tema, el que vol dir que ha d'estar dissenyat d'una manera simple perquè sigui fàcil d'estudiar i entendre. Tot i ser el més simple possible, ha de complir tots els requeriments perquè sigui el més complet i educatiu possible.

Costos

- Un altre requeriment que hem imposat a l'hora de desenvolupar aquest projecte, ha estat el de fer ús en mesura del possible de materials que estiguin disponibles al departament i que es puguin utilitzar.
- Relacionat amb l'apartat anterior, s'ha mirat que els costos de desenvolupament siguin els menors possibles, per tal de tenir una eina d'aprenentatge útil, completa i de baix cost.

¹ PBL (Program Based Learning). Mètode d'aprenentatge basat en problemes.

Al principi d'un projecte pots fer molts càlculs de com es durà a terme, fer estimacions de dates previstes, calcular recursos,....., però la veritat és que mai es pot estar segur d'aquests càlculs. I és això precisament el que ha passat amb aquest projecte. Al principi es va fer una estimació aproximada sobre com podien anar les coses.

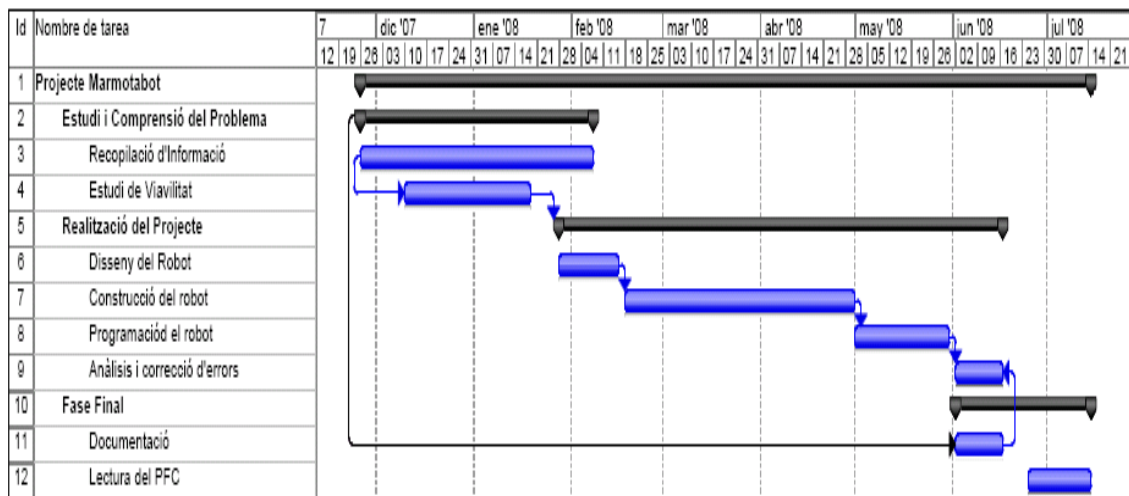
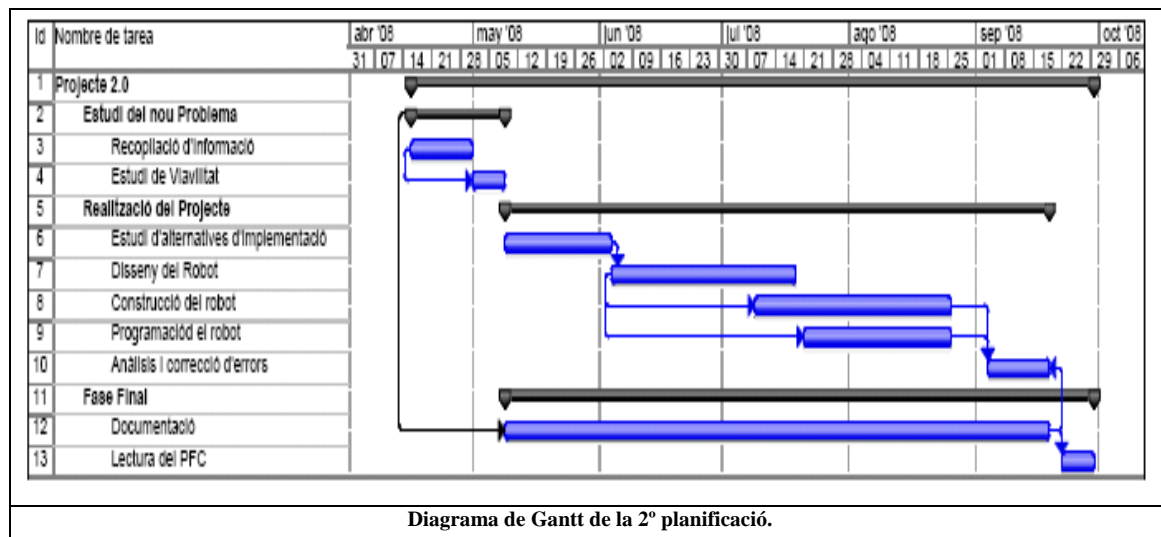


Diagrama de Gantt de la 1ª planificació.

El diagrama que es mostra en el quadre superior representa l'estimació i planificació que es va fer en un primer moment a l'hora de començar el projecte. Com es pot veure el projecte estava planificat per arribar al seu final en juny. Això no va ser possible degut a molts factors, com poden ser canvis en l'estructura i l'objectiu del projecte, retards en alguns aspectes del projecte, exàmens i altres aspectes acadèmics,.... Aquests entre d'altres motius van derivar en l'aplaçament de l'entrega, al setembre.

Degut a tots aquests canvis i decisions, es va fer una nova planificació del projecte



Aquesta segona planificació s'ha mirat de seguir el més fidelment possible quan les circumstàncies ho permetien. Tot i això en moltes ocasions s'han hagut d'ampliar els límits de tems d'algunes tasques, i d'altres han començat i acabat abans del que es tenia previst.

Un cop hem vist els requisits d'aquest projecte, és hora de començar a buscar solucions d'implementació que s'adaptin al que estem buscant.

A l'estiu de 2005 Parallax Inc. va introduir al mercat el robot d'aprenentatge i formació SCRIBBLER, que és una de les opcions més completes i econòmiques del moment per introduir-se en l'aprenentatge de la robòtica.

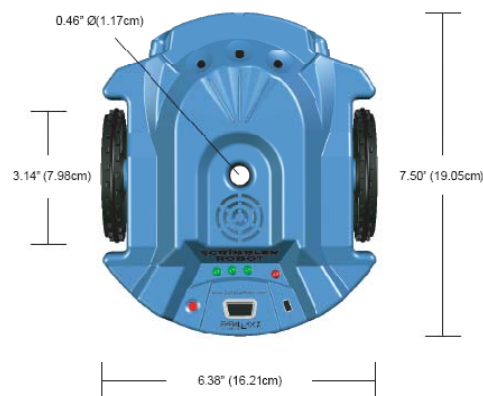
Una de les característiques més destacables del robot és el seu senzill disseny i el fet que ve totalment muntat de fàbrica, incloent diversos sensors, motors i complements que permeten dur a terme nombroses funcions i aplicacions. Aquesta característica fa que moltes persones que no podrien iniciar-se en la programació i la robòtica, degut als seus nuls coneixements d'electrònica i mecànica, puguin donar els seus primers passos en la programació de robots. Un altre de les seves avantatjoses característiques per l'usuari comú, són els programadors que es faciliten a l'hora de comprar el robot. Tenim dues maneres de programar el robot, de manera fàcil i intuïtiva utilitzant el programa scribbler GUI, en el qual no es precisa cap coneixement previ o bé amb l'editor de codi PBasic.

Sembla ser que pot ser una bona opció com a model de referència. A continuació veurem amb més detall les característiques d'aquest robot.

Hardware

El robot bé muntat de fabrica amb un xassís de plàstic on estan posicionats els sensors i connectors. Al seu interior trobem el microcontrolador. Veiem més a fons les característiques hardware del sistema.

Mides y connexions



- Disposa d'un port sèrie per a la programació del robot.
- També està disponible un adaptador per a la connexió mitjançant bluetooth.
- Botó de Reset, per esborrar la programació del robot.

Sensors

- 2 emissors infraroigs a cada costat de la cara frontal i un receptor Infraroig en mig de la cara frontal.
- Tres sensors de llum (fotoresistències) muntades a la part frontal superior.
- 1 altaveu per a generar sons i notes musicals
- 3 LEDs programables
- Un parell d'emissors i receptors d'infraroigs muntats a la base i encarats a terra per a la detecció de línies.
- Dos motors programables (DC, PWM)
- Un sensor de protecció de motors (stall sensing capability)
-

Placa Controladora

- Tarja controladora BASIC Stamp2

Velocitat del processador:	20MHz
----------------------------	-------

Velocitat d'execució:	~4000 instruccions per segon
RAM:	32 Bytes (6 per I/O, 26 variables)
EEPROM:	2K Bytes, o ~500 instruccions
I/O pins:	16 (+ 2 dedicades)
Comandes PBASIC:	42

Alimentació

- 1 porta-piles para 6 piles AA de 1,5 V alcalines, normals ó de Ni-MH.



Software

El robot scribbler es subministra acompanyat d'un software de programació, que permet dissenyar el comportament del robot de diverses maneres, ja sigui mitjançant l'editor gràfic, o bé utilitzant codi PBASIC en l'editor de text.

El robot scribbler es subministra amb un editor gràfic molt intuïtiu i fàcil d'utilitzar per als més petits.

Mitjançant la interfície gràfica podem implementar la funcionalitat del robot d'una manera molt fàcil. Aquesta interfície consisteix en una finestra del PC on se'ns dona l'opció d'escollir entre diferents blocs de colors, on cada bloc representa una possible acció que pot dur a terme el robot. Al costat tenim una altra finestra on hem d'anar arrossegant els blocs esmentats, organitzant-los en ordre segons la funcionalitat desitjada. A l'hora d'escollir cada bloc se'ns obre una finestra per especificar les variables i opcions per a cada acció del robot.

Amb robot scribbler es subministra també un editor de tex per als usuaris una mica més avançats que vulguin programar la funcionalitat del robot mitjançant línies de codi.

La programació es fa en el llenguatge d'alt nivell PBASIC, i no són necessaris massa coneixement per començar a dissenyar programes senzills per al robot. Simplement hem de començar a escriure el codi amb la funcionalitat del robot.

Programador Gràfic



Scribbler Program Maker GUI

Programador de Text



BASIC Stamp Editor

El model del robot Scribbler és molt adequat per a nens o joves que estan donant els seus primers passos en el món de la robòtica i que no disposen de grans coneixements en aquest àmbit. Pel que fa al l'usuari al que va destinat el projecte, al estudiar el model Scribbler més a fons trobem una sèrie traves i limitacions que condicionen molt l'ús del robot i limiten en gran manera l'estudi i la docència de determinats temes de l'àmbit de la informàtica. A continuació veurem algunes d'aquestes limitacions.

- Estructura del robot. Com hem vist abans el robot Scribbler ve encapsulat de fàbrica cosa que impedeix l'expansió del robot pel que fa als sensors o canvi de determinades parts, limitant-nos als que incorpora, els quals són pocs i limiten l'experiència docent. Encara que incorpora, com hem vist anteriorment, un “*hacker port*” per expandir algunes funcionalitats, aquest és completament insuficient per als nostres propòsits.
- Programació d'alt nivell. Amb aquest model només se'ns permet una programació d'alt nivell, en PBasic, limitant a aquest àmbit tot el referent a la programació. Aquesta característica és un altre inconvenient, ja que per tal d'integrar el sistema en el simulador pyrobot², que està escrit en python³, convindria poder treballa amb un altre llenguatge d'alt nivell com pot esser el C.
- Programació de baix nivell. El model no permet cap tipus de programació de baix nivell. Degut als requisits i a l'aplicació al qual està destinat necessitem la possibilitat de programar a baix nivell en llenguatge assemblador.

² Pyrobot simulator. Entorn de simulació de comportaments dins els software Pyro de Python Robotics.

³ Python. Llenguatge de programació d'alt nivell, semblant al Perl.

- Inviabilitat per a un PBL. Degut a totes aquestes característiques i al fet de l'arquitectura del microcontrolador és complicada, cosa que no afavoreix la facilitat en l'aprenentatge, podem dir que aquest model de robot no ens serveix pels nostres propòsits.

Com podem veure, aquest model presenta una sèrie d'inconvenients que el fan inviable per aconseguir els nostres objectius. Per aquest motiu ens veiem obligats a buscar alternatives de disseny per la construcció de la plataforma robòtica.

Com acabem de veure, el robot scribbler és una bona solució per iniciar-se en el món de la programació de robots, partint d'una base quasi bé nul·la de coneixements. Això canvia quan volem progressar en l'estudi i l'aprenentatge de certs aspectes de la matèria, essent l' scribbler una limitada solució.

Per tal d'aconseguir una eina de treball i aprenentatge més completa i que ens permeti implementar les funcionalitats desitjades, superant d'aquesta manera les limitacions del scribbler, hem de pensar en algunes alternatives que compleixin els requisits establerts.

A continuació es mostren algunes possibles alternatives d'implementació per tal de corregir les deficiències de l'antic model:

Utilització de "Hacker Port"

Per solucionar el problema de l'expansió de funcionalitat del robot podem fer ús del "hacker port" que el model scribbler incorpora.

Això ens permetria el disseny i la connexió d'un conjunt de circuits addicionals, permetent augmentar la funcionalitat del robot. S'ha de tenir en compte que el disseny de la nova circuiteria, s'adapti a l'estructura del robot, per tal de no restar funcionalitats.

Tot i que podem fer ús d'aquest "hacker port" el número de línies és totalment insuficient, i d'altra banda aquesta proposta no soluciona l'inconvenient de l'impossibilitat de la programació en assembleador.

Dins la gama de productes que ens ofereix la casa Parallax, casa a al qual pertany el model Scribbler, trobem un altre model de robot educatiu anomenat Boebot. A diferència del Scribbler, el Boebot és un robot que es subministra per peces totalment desmuntat. És treball de l'usuari el muntatge i programació del robot. Aquesta característica soluciona el problema de l'expansió del robot i la possibilitat de substituir peces per afegir o eliminar funcionalitats del robot. També seria una bona experiència educativa per a usuaris més avançats que tinguin coneixements d'electrònica i robòtica, que siguin capaços de muntar el robot.

Tot i aquests avantatges, el robot Boebot, a l'igual que el robot Scribbler, treballa amb un microcontrolador Basic Stamp 2, cosa que no soluciona el problema de la programació a baix nivell, en ensamblador, ni la programació d'alt nivell hauria de ser en PBasic, impossibilitant el treball en C o Python.

Per altra banda tindríem el problema que les peces van situades a sobre el robot, sobre una protoboard. Essent el robot una eina d'aprenentatge que passaria per les mans de molts usuaris, és del tot probable que s'hagués d'estar muntant el robot a cada moment per la caiguda de peces, cosa que donaria molts problemes i seria una pèrdua de temps.

Cross compiler – Ensamblador → PBasic

Per intentar solucionar l'inconvenient de la nul·la possibilitat de programació en ensamblador per a l'usuari del robot scribbler, es pot pensar en una solució alternativa per realitzar aquesta tasca. Aquesta solució passaria per crear un compilador creuat de codi ensamblador a Pbasic, que és el llenguatge que reconeix el microcontrolador.

Amb aquest mètode tindríem la possibilitat de programar el robot en ensamblador. L'inconvenient resideix en el fet que no estem programant en ensamblador realment i per tant ens és impossible veure i aprendre a fons l'ús de registres i altres unitats de control a més baix nivell, essent aquest un dels objectius que ha de complir el robot.

Recrear Scribbler amb un Microcontrolador

Pensant més alternatives als problemes que trobem al model Scribbler, ens trobem amb un altre alternativa una mica més elaborada. Aquesta alternativa consisteix en la construcció d'un robot a semblança del Scribbler, amb les seves funcions i sensors, però amb la característica que canviem el Basic Stamp per un altre microcontrolador que permeti les funcionalitats desitjades i faciliti el treball amb el nou model. Aquest microcontrolador, que pot ser un PIC, AVR, ..., és més potent i té més i millors característiques que el Basic Stamp 2, fent possible d'aquesta manera la implementació de noves funcionalitats.

Aquesta solució és més complexa que les altres i s'han de tenir en compte els costos dels materials i de construcció del nou robot. Per altra banda, el nou microcontrolador ens dóna molta més llibertat de treball amb el robot, ja que és més potent, més ràpid, té més E/S i altres característiques que facilitaran la construcció del robot.

Elecció de l'alternativa

Un cop vistes les diferents alternatives que podem dur a terme. S'ha decidit optar per l'última, és a dir, el re-disseny del robot scribbler substituint el microcontrolador BasicStamp per un altre més potent i que permeti un augment de les possibilitats del robot. Aquesta alternativa inclou tots els requisits vistos anteriorment, i soluciona els problemes vistos anteriorment.

El microcontrolador escollit es podrà veure mes endavant en la secció de la Unitat de control.

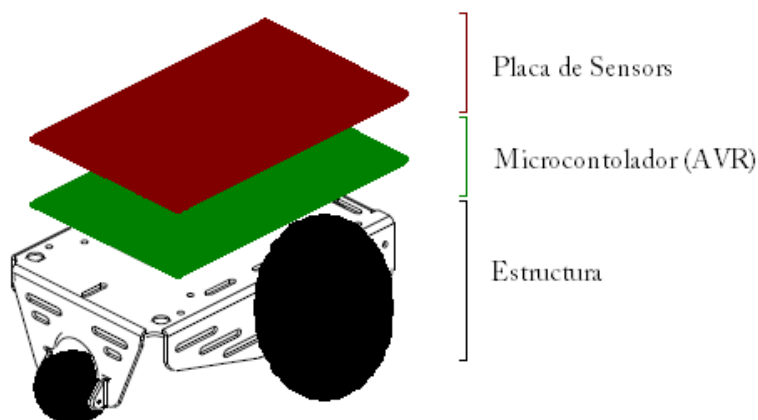
Model Físic

Un cop hem fet l'elecció de l'alternativa més recomanable i completa, entrem de ple en la fase de disseny e implementació d'aquesta alternativa. En els següents apartats explicarem el disseny que s'ha creat, veient les diferents parts i les característiques de cada component.

Estructura

L'estructura final del robot esta determinada pels materials disponibles emprats en la construcció del mateix, amb l'única restricció que el model dissenyat sigui d'una mida que permeti una còmoda manipulació del robot.

En principi l'estructura bàsica del robot anirà determinada pel xassís del robot boebot de la companyia Parallax. Principalment s'ha escollit aquesta opció per tal de reutilitzar materials disponibles i reduir el màxim possible els costos addicionals. Tot i això, degut a la seva senzillesa i versatilitat, l'estructura s'adapta perfectament a les necessitats estructurals del robot dissenyat.



Estructura bàsica del Robot

El robot està suportat per dues rodes motrius, que impulsaran el robot, i una roda neutra que donarà estabilitat i suport a l'estructura.

Aquestes rodes aniran impulsades per dos petits servomotors de 5V, amb potència suficient per desplaçar sense problemes tota l'estructura. Els servomotors estaran controlats directament per la placa microcontroladora de manera totalment independent, permetent d'aquesta manera la capacitat de realitzar girs sobre l'eix del robot. Aquesta manera de control dels motors ens permetrà una àmplia varietat de moviments, fent possible el moviment i la navegació del robot.

Alimentació

En el referent a l'alimentació necessària pel correcte funcionament del robot, hem de tenir en compte l'abastiment suficient en dos aspectes bàsics. Els dos referents de consum d'energia en el robot seran els servomotors i la placa microcontroladora.

Per subministrar l'alimentació necessària per al funcionament del robot, farem ús d'un porta piles amb capacitat de 4 piles de tipus AA de 1'5V o d'un porta piles d'una pila quadrada de 9 V. El porta piles anirà situat a la part inferior del xassís per tal situar el centre de gravetat el més avall possible i donar més estabilitat al conjunt estructural del robot.

Control

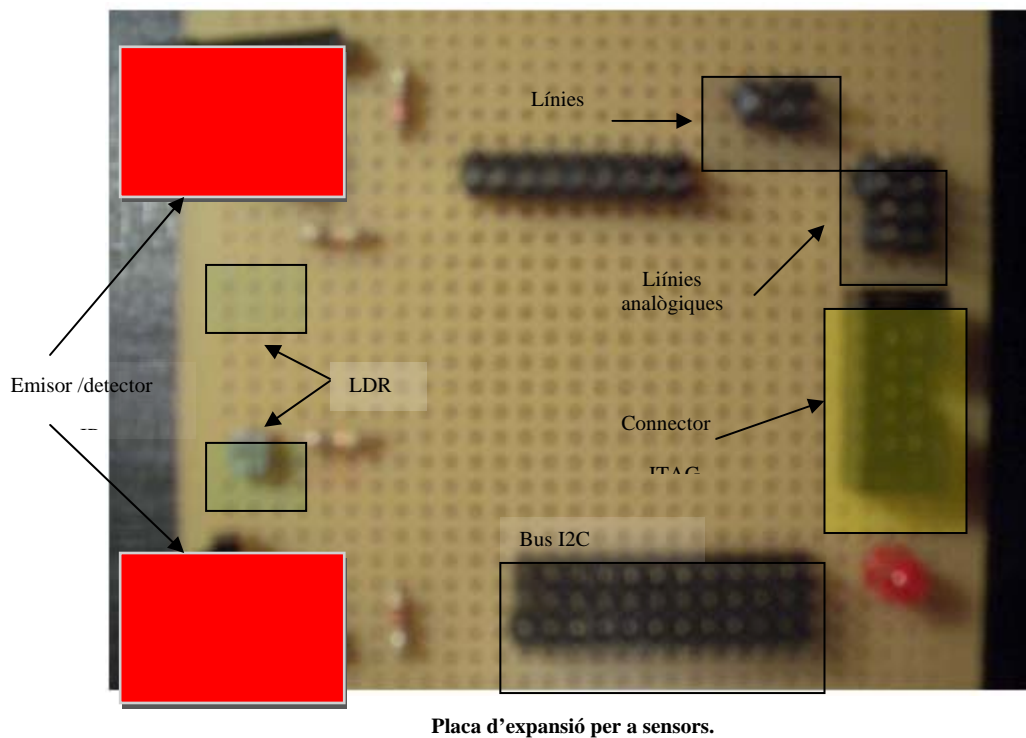
El control del robot estarà governat pel que podríem anomenar "unitat de control", que gestionarà tant els motors, com els diferents sensors dels que disposa el robot.

Bàsicament consta d'un microcontrolador AVR ATmega 128 de la companyia ATMEL. Aquest microcontrolador incorpora totes les funcions i característiques

necessàries pel total control del robot. Veurem aquestes característiques en detall més endavant.

Sensors

El robot disposa de diferents sensors i actuadors per tal de dotar-lo de certes funcionalitats, permetent a l'usuari l'aprenentatge i la programació del funcionament d'aquests sensors. Els sensors incorporats són de diferent tipus, com sensors infraroigs, lumínics, Es disposa de ports d'expansió per donar la possibilitat d'afegir nous sensors. Els diferents sensors van situats en una petita placa d'expansió, muntada a sobre del robot.



La incorporació d'aquests sensors obre un ventall de possibilitats i funcionalitats que permeten a l'usuari una millor experiència de programació i fa més entretingut i amè el procés d'aprenentatge.

La unitat de control serà l'encarregada de gestionar els programes o comportaments del robot. Totes les operacions de càlcul i execució les durà a terme la U.C, encarregant-se també de l'execució dels programes i la presa de decisions del robot. Podríem dir, que d'alguna manera la unitat de control serà l'ànima i la intel·ligència del robot.

Pel que fa al nostre projecte, tenim a la nostra disposició un gran ventall de microcontroladors per tal d'implementar la unitat de control del nostre robot. Com ja hem dit anteriorment, les tecnologies avancen molt ràpidament i aquestes plaques ens ofereixen una quantitat de recursos i una potència operativa molt elevades, cosa que ens permet fer un disseny més potent i operatiu que el nostre model de referència.

A continuació veurem diferents alternatives d'implementació, en el referent al microcontrolador, on es mostren les principals característiques i peculiaritats d'algunes famílies de microcontroladors de 8-bits. Bàsicament veurem algunes de les marques comercials més importants i més esteses, com són Atmel (AVR), Microchip Tech. (PIC) i Freescale (Motorola).

AVR (Atmel)

Els AVR són una família de microcontroladors RISC, de la casa Atmel, basats en una arquitectura Harvard. Els AVR van ser una de les primeres famílies de microcontroladors que van incorporar memòria Flash per l'emmagatzematge de programes, utilitzant la memòria EEPROM per les dades.

A més a més, aquesta arquitectura va ésser dissenyada per ser totalment compatible amb el llenguatge C, permetent treballar a més alt nivell deixant la tasca d'optimització del codi objecte a l'assemblador, augmentant d'aquesta manera l'eficiència a la programació.

Disposa també d'un complet i potent joc d'instruccions que s'executen en un sol cicle de rellotge.

Arquitectura

La família de microcontroladors AVR, és considerablement extensa i tots els models comparteixen el mateix nucli AVR, però tenen diferents perifèrics, i quantitats de RAM i ROM. Es basen en una arquitectura Harvard, amb busos i memòries separades per instruccions i dades (EEPROM - Instruccions, RAM - dades). La CPU és de tipus RISC⁴. Tot això afavoreix la velocitat del sistema.

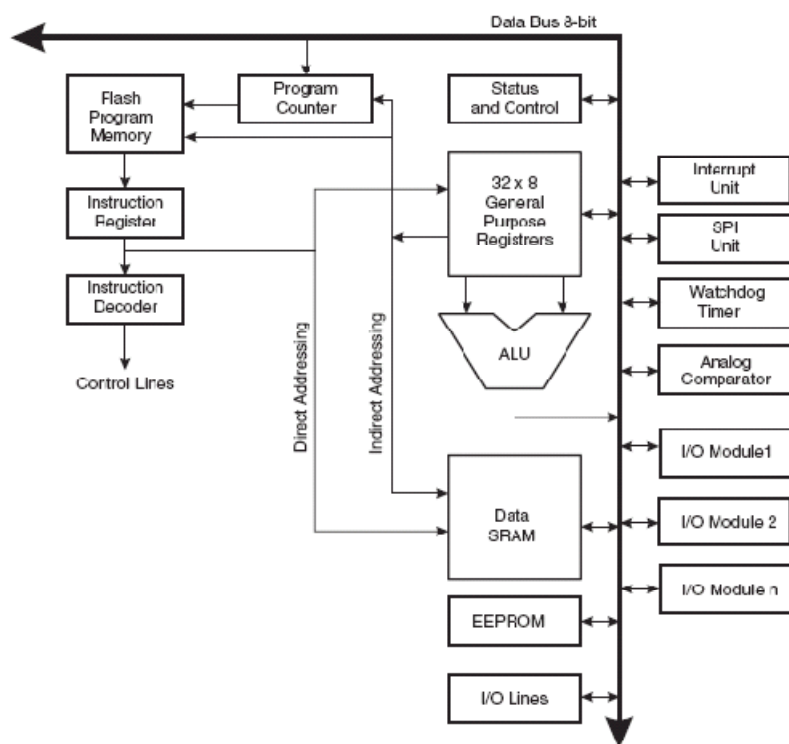


Diagrama de blocs de la arquitectura AVR

Com podem veure en la imatge superior, els 32 registres de treball de propòsit general estan directament connectats a la Unitat Aritmetico-Lògica (ALU), permetent que dos registres independents s'introdueixin en una sola instrucció executada en un cicle de rellotge. Les instruccions que estan en la memòria del programa s'executen amb

⁴ RISC : Reduced Instruction Set Computer.

un sol nivell de transmissió per conductes. A la vegada que s'executa una instrucció, s'extreu la següent instrucció de la memòria del programa. Aquest concepte permet que les instruccions s'executin en un sol cicle de rellotge.

Eines de desenvolupament

Com hem dit l'AVR està força estesa en el món dels microcontroladors, el que ofereix una extensa gama de programadors i software per treballar amb el μ C.

L'AVR disposa d'un gran ventall d'eines de programació. Podem trobar gran varietat de programadors per a diferents games de microcontroladors, així com diferents mètodes de connexió entre el microcontrolador i el PC. Per exemple entre d'altres podem trobar programadors per ISP, JTAG, ..., amb diferents mètodes de connexió de connexió com poden ser paral·lel, sèrie o USB.

AVR Studio és un Entorn de Desenvolupament Integrat (IDE) professional per escriure i “debugar” aplicacions AVR en Windows. AVR Studio és una eina de desenvolupament totalment gratuïta que ens permetrà programar i “debugar”, també “in circuit”, els programes creats pels microcontroladors AVR. AVR Studio suporta llenguatge de baix nivell en assemblador.

WINAVR és un programa compilador de C. És totalment gratuït i un cop instal·lat es pot incorporar com a part de l'AVR Studio permetent crear i compilar programes en C.

PIC (Microchip Technology)

Els PICs són una família de microcontroladors RISC, basats en una arquitectura Harvard, fabricats per la companyia Microchip Technology Inc. Els PICs deriven del

PIC1640, originàriament desenvolupat per la divisió de microelectrònica de General Instruments.

Els PICs són molt populars en el món de la robòtica i la microelectrònica, degut a el seu reduït cost, la seva versatilitat, la gran col·lecció d'aplicacions i les seves eines de desenvolupament gratuïtes.

Arquitectura

La família de microcontroladors PIC, és una de les famílies de microcontroladors més expandides entre la comunitat de microcontroladors, amb una ampla gama de microcontroladors, trobant diferents games, amb models que es diferencien en capacitats de memòria, quantitat de perifèrics,..... Es basen en una arquitectura Harvard, amb busos i memòries separades per instruccions i dades, memòria EEPROM per instruccions i memòria RAM per les dades. La CPU és de tipus RISC.

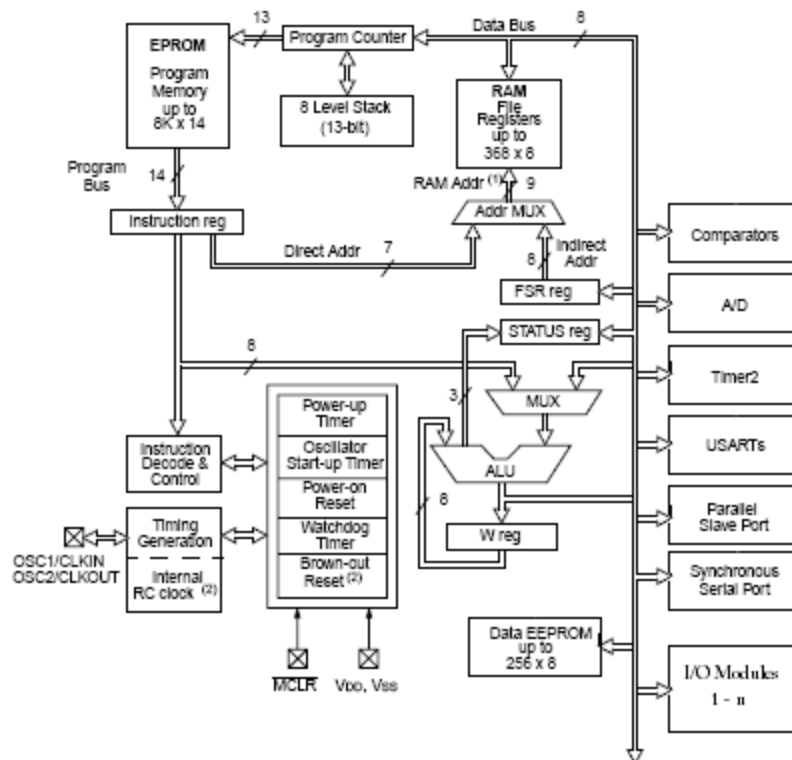


Diagrama de blocs de la arquitectura PIC de gama mitja

En el diagrama anterior de l'arquitectura del PIC, podem veure que la Unitat Aritmetico-Lògica utilitza un registre acumulador W (Working, no direccionable), i el resultat es posa directament al bus de dades. La configuració de la Unitat Aritmetico-Lògica és una mica més complexa, carregant registres abans de fer les operacions, i trigant normalment 3 o 4 cicles de rellotge.

Eines de desenvolupament

Degut que els PICs són àmpliament utilitzats en la comunitat d'aficionats a la robòtica, existeixen diferents i molt variats programadors amb els que podem treballar. Es pot trobar a la xarxa un nombre molt elevat de dissenys de circuits, layouts i esquemàtics que permeten construir un simple programador propi per començar a treballar amb el PIC.

El programador U2 de Melabs es poden programar fàcilment la majoria de microcontroladors PIC. Serveix tant per programar "In Circuit", com mitjançant un adaptador ZIF disponible per encapsulats de 8, 14, 18, 20, 28 i 40 terminals

MPLAB és un entorn de desenvolupament molt complet que es pot descarregar de manera totalment gratuïta des de la pàgina web de Microchip. MPLAB incorpora l'assemblador MPASM, per compilar el llenguatge assemblador de baix nivell. També incorpora una demo del compilador de C de l'empresa CCS, encara que és una versió molt limitada. Per la simulació disposa de manera gratuïta del simulador MPSIM.

FREESCALE (MOTOROLA)

Els microcontroladors de la companyia Freescale, divisió independent de Motorola, són molt coneguts i utilitzats, ja que són l'evolució dels microcontroladors de Motorola, que eren molt fiables. Dins de la varietat de microcontroladors de propòsit general de Freescale, podem trobar alguns molt interessants. Tot i el seu estès ús, no compta amb una comunitat tan àmplia com les anteriors marques, cosa que fa que no hi hagi tantes eines per treballar amb aquests microcontroladors.

La família Mc68Hc11 de Freescale té una arquitectura harvard, però contràriament el que passava amb els models anteriors la CPU és CISC⁵, un joc d'instruccions complexes que s'executen en mes d'un cicle de treball. Podem veure en les línies inferiors un diagrama de l'arquitectura d'aquest microcontrolador.

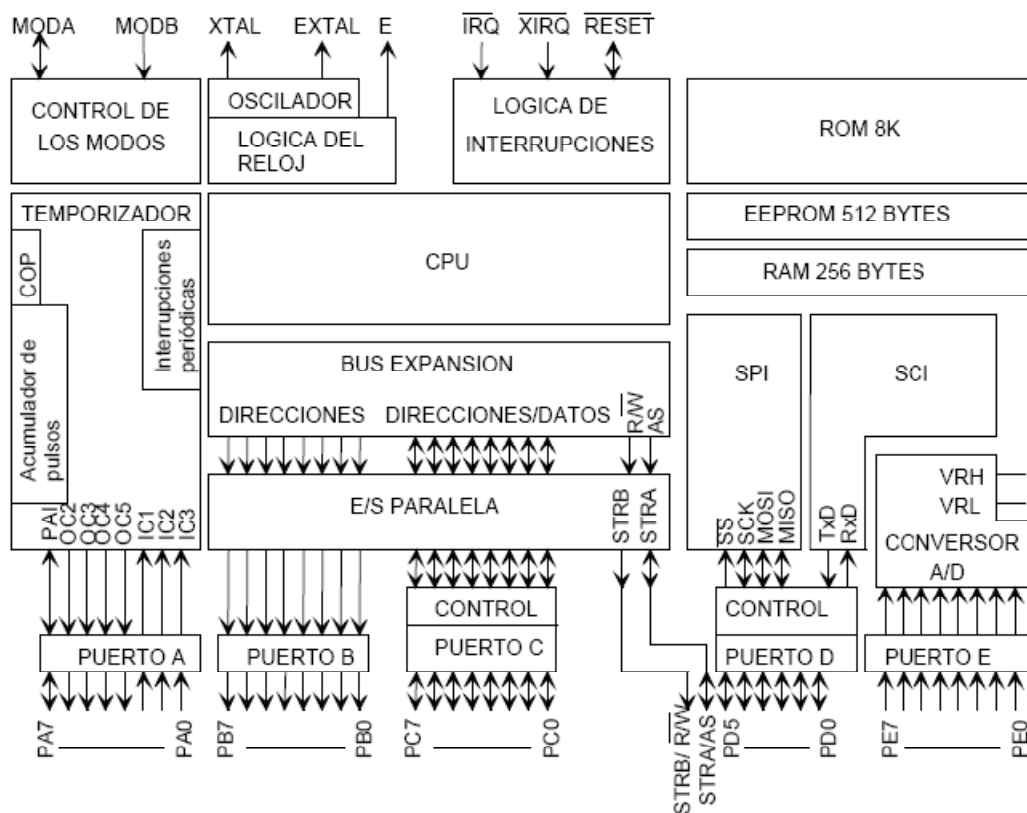


Diagrama de blocs de la arquitectura de la família MC68HC11 de Freescale.

En el diagrama anterior podem veure els diferents mòduls de memòria on es dipositen les dades i les instruccions. També veiem la distribució de mòduls del microcontrolador.

⁵ CISC: Complex Instruction Set Computer.

Per fer ús dels microcontroladors, Freescale posa a disposició de l'usuari algunes alternatives de programació, entre d'elles el CodeWarrior, un entorn de desenvolupament per a microcontroladors. La versió completa del Software CodeWarrior és de pagament, encara que si ens registrem a la pagina de Freescale, podrem descarregar una versió d'avaluació.

Exemples

A les taules es mostra algunes característiques de les famílies de microcontroladors estudiades.

	Flash (KBytes)	EEPROM (Kbytes)	SRAM (KBytes)	I/O Pins	Freq. Max (MHz)	A/D Channels
Lighting AVR	8 - 16	0.5	0.5 - 1	19 - 27	16	--
Tiny AVR	1 - 8	0.125 - 0.5	0.125 - 0.5	6 - 24	4 - 20	4 - 11
Mega AVR	4 - 128	0.25 - 4	0.5 - 8	23 - 86	16 - 20	8 - 16

Famílies de 8-bits de Atmel

	Flash (KBytes)	EEPROM (Bytes)	RAM (Bytes)	I/O Pins	CPU Speed (MHz)	A/D Channels
PIC16 Family	0.75 - 14	0 - 256	24 - 368	12 - 54	20 - 40	0 - 14
PIC18 Family	4 - 128	0 - 1024	256 - 3968	15 - 70	32 - 64	4 - 16

Famílies de 8-bits de Microchip

	Mask ROM (KBytes)	Flash (Bytes)	EEPROM (Bytes)	Int. RAM (Byte)	I/O Pins	Bus Freq. Max (MHz)	A/D Channels
68HC05	1 - 32	--	76 - 8192	64 - 920	10 - 39	1 - 4	4 - 8
68HC08	0 - 312	0 - 60	0 - 2024	128 - 2048	21 - 64	4 - 8.4	4 - 24
68HC11	4 - 32	--	512 - 640	192 - 1000	16 - 51	2-5	0 - 8

Famílies de 8-bits de Freescale

Després d'aquest petit estudi dels diferents microcontrolador que podem trobar, hem de dir que cada companyia té una varietat molt elevada de microcontroladors, és possible trobar un microcontrolador específic que s'adapti a cada projecte que es desitgi fer, microcontroladors amb més o menys memòria, entrada/sortida, recursos, velocitat de procés,

Una característica que s'ha de tenir en compte a l'hora d'escollir el microcontrolador és la seva arquitectura i estructura. Els μC de Microchip i Freescale disposen d'una arquitectura més complexa, on una instrucció triga més d'un cicle d'execució, l'arquitectura dels μC s Atmel pel contrari, és més simple i entenedora, i les instruccions s'executen en un sol cicle.

Un altre aspecte important són les eines de desenvolupament disponibles, ja que són el mitjà que disposem per tal de treballar amb el microcontrolador. Quant més fàcil i entenedor, al temps que complet, sigui l'entorn de desenvolupament del μC , més fàcil serà el seu aprenentatge i ús.

Elecció

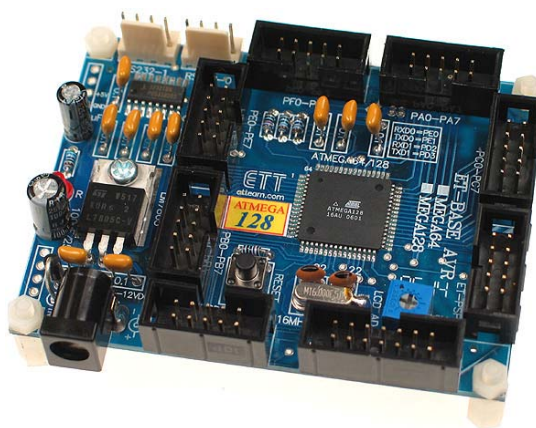
En el cas d'aquest projecte, gràcies als recursos disponibles per la realització del projecte, es va poder utilitzar un microcontrolador AVR Atmega128 de la companyia Atmel. Es va estudiar les característiques del μC per tal d'intentar incorporar-lo al projecte. Finalment aquest μC de 8-bits oferia tots els recursos necessaris per a la implementació del projecte:

- L'arquitectura és prou elegant i entenedora, per poder utilitzar-la com a model d'ensenyament en la introducció als processadors.
- Gran capacitat de programació, a baix nivell (assemblador) i alt nivell (C i altres) que permeten un ampli espectre de programació.

- El baix cost del microcontrolador, i el fet que estigui muntat sobre una placa d'expansió per facilitar la connectivitat.
- Gran varietat i qualitat de recursos de programació i simulació al nostre abast
- Un μC potent (16MHz) amb gran capacitat d'entrada/sortida (56) i multitud de recursos a la nostre disposició (Timers, A/D, PWM, ...)

El model de microcontrolador escollit és el ATmega128, que com s'ha vist abans, és una maquina Harvard tipus RISC amb 128 Kbytes de memòria Flash per instruccions i 4 Kbytes de memòria RAM, disposa de multitud de dispositius integrats, com poden ser un bus I2C, 8 canals de conversor A/D, quatre timers, dos connectors RS232, generador de PWM, gran quantitat de connectors E/S, entre moltes altres característiques.

La placa ET-ATmega128 és una placa de propòsit general, fabricada per la companyia tailandesa ETT. El microcontrolador que és de tipus SMD⁶ ve muntat ja a la placa cosa que facilita la seva utilització. La placa (que es pot veure a la imatge), és bàsicament el microcontrolador i l'expansió dels pins d'aquest. Aquesta expansió permet treballar de manera més còmoda i dóna fàcil accés a les línies del microcontrolador.



Placa ET-Base AVR ATmega128

El port F incorpora els pins que permeten programar el microcontrolador mitjançant el programador JTAG. També incorpora un connector per la programació amb un programador ISP.

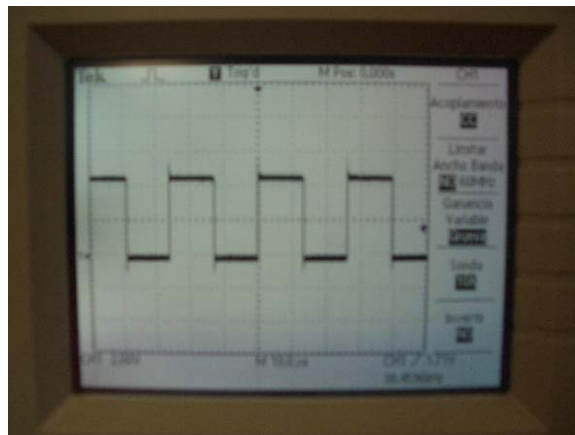
Per més informació veure bibliografia per trobar el datasheet.

⁶ **Surface Mount Device**, és un tipus d'encapsulat pensat per soldar a sobre de la placa sense necessitat de perforat. Esta pensat per ser soldat a màquina.

S'han incorporat alguns sensors per tal de poder millorar l'experiència de la programació. Aquests sensors van distribuïts a sobre d'una placa d'expansió per fer més fàcil l'enteniment de la connectivitat. Els sensors afegeixen al robot la possibilitat d'un cert nivell d'interacció amb el món real mitjançant la captura de dades de l'entorn. Alguns dels sensors incorporats són infraroigs per la detecció d'obstacles i el seguiment de línies, LDR o fotoresistències per la detecció de fonts de llum, i un compàs magnètic el posicionament del robot. També s'han creat els drivers necessaris per tal de poder començar a utilitzar aquests sensors de manera ràpida i senzilla.

Infraroigs

Els infraroigs funcionen mitjançant l'emissió, per part d'un díode emissor infraroig, de senyals a una certa freqüència. Aquestes reboten en els diferents objectes que troben al seu pas, sent captades per un receptor d'infraroigs. Aquest receptor, està calibrat per detectar senyals a una determinada freqüència i quan detecta el senyal de sortida canvia.



Senyal cuadrada de 38 Khz pels infraroigs

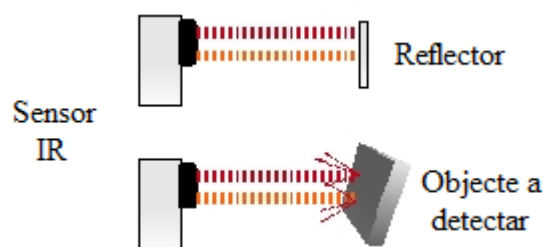
Els rajos infraroigs (IR) entren dins del fototransistor on trobem material piroelèctric, natural o artificial. S'ha de tenir present que aquesta configuració és sensible a la llum ambient perjudicant les mesures, poden donar lloc a errors, hem

d'incorporar circuits de filtratge en termes de longitud d'ona, per tant serà important que treballin en ambients de llum controlada.

El robot disposa de dos jocs diferents de sensors infraroigs. Cada un està dissenyat amb unes característiques específiques el que ens serveix per aplicar-ho a diferents aspectes de control del robot.

Detector obstacles

En primer lloc tenim el sistema de detecció d'obstacles que consta de dos díodes emissors de llum infraroja i dos receptors infraroigs PNA4601M de la casa Panasonic. Els receptors estan calibrats per tenir una recepció òptima de senyals de freqüències al voltant dels 38.5 KHz. Aquest receptor està indicat per la detecció d'obstacles, ja que té un rang de detecció força ampli, el que dona temps al robot a dur a terme les accions necessàries.



Principi de funcionament del IR

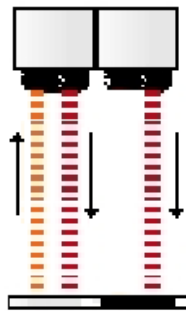
Per més informació, es poden veure les característiques tècniques en l'apèndix I.

Seguidor de línies

El sistema de seguiment de línies està compost per dos sensors GP2Y0D805Z0F, dissenyats per la companyia Sharp. Aquests sensors disposen d'emissor i receptor incorporats els que facilita el seu ús per l'aplicació requerida. La distància de recepció és força més curta que en el cas anterior el que serà un avantatge pel seu ús en el rastrejador de línies. Un aspecte a tenir en compte en aquests tipus d'ús

del sensor IR, és el coeficient de reflectivitat dels objectes, no per totes les superfícies el sensor treballarà igual.

El funcionament del seguidor de línies és el següent. Disposem de dos sensors IR de curta distància, que aniran encarats a la superfície on està la línia a seguir. Les propietats de reflectivitat dels diferents colors i superfícies, ens serviran per controlar la posició respecte a la línia. Un sensor detectarà línia, i l'altre no. Amb això gestionarem el seguiment.



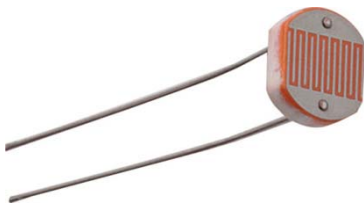
Funcionament bàsic del seguiment de línies.

Per més informació, es poden veure les característiques tècniques en l'apèndix I.

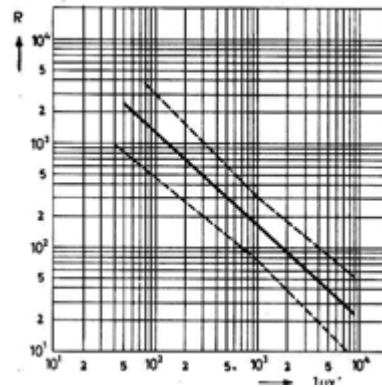
Fotoresistència (LDR)

LDR o Light-Dependent Resistor és una resistència que varia el seu valor de resistència en funció de la quantitat de llum que incideix sobre ell. El valor de resistència elèctrica d'un LDR és baix quan la llum incideixen ell. En el cas contrari en condicions de foscor el valor de resistència és molt eleva. Les LDR utilitzades són de la sèrie VT900 i seran utilitzades pel sistema detector de llum.

Les LDR es basen en la variació de la resistència elèctrica d'un semiconductor al incidir sobre ell una radiació òptica (entre 1mm i 10 nm) que aporta energia per augmentar el nombre d'electrons lliures (efecte fotoelèctric) fent disminuir la resistivitat del material. La sensibilitat a les diferents longituds d'ona, vindrà determinada pel material que esta fabricada la fotoresistència.



Fotoresistència o LDR



Característiques resistència-il·luminació en un LDR

Per més informació, es poden veure les característiques tècniques en l'apèndix I.

Compàs Magnètic

Per afegir orientació geogràfica al robot, hem utilitzat un mòdul CMPS03 de la casa Devantech. El compàs magnètic o brúixola digital, és un dispositiu electrònic que determina un angle respecte al camp magnètic de la Terra. Utilitza dos sensors magnètics situats en diferents eixos, que son suficientment sensibles com per detectar el camp magnètic terrestre. Aquest compàs magnètic té una resolució de 0.1 graus i una precisió de 3 a 4 graus.



Compàs Magnètic CMPS03 de Devantech.

Per més informació, es poden veure les característiques tècniques en l'apèndix I.

Els servomotors o també anomenats servos, son una classe especial de motors de corrent continua, que es caracteritzen per la seva capacitat de posicionar-se de manera immediata a qualsevol posició del seu rang d'operació. Degut a aquesta característica de posicionament, són molt populars i utilitzats en el món del modelisme, i altres hobbies de radio control.

Encara que hem inclòs els servomotors dins la secció de sensors, s'ha de dir que el servomotor no és un dispositiu d'entrada de recollida de dades, sinó que un cop rebudes algunes dades, ens permet actuar en conseqüència.

El robot dissenyat incorpora dos servomotors per tal de fer possible el moviment del robot. Aquests servomotors estan situats a la base de l'estructura del robot, posicionats com dispositius de locomoció. Els servos utilitzats són de la casa Parallax i són servomotors de rotació continua.



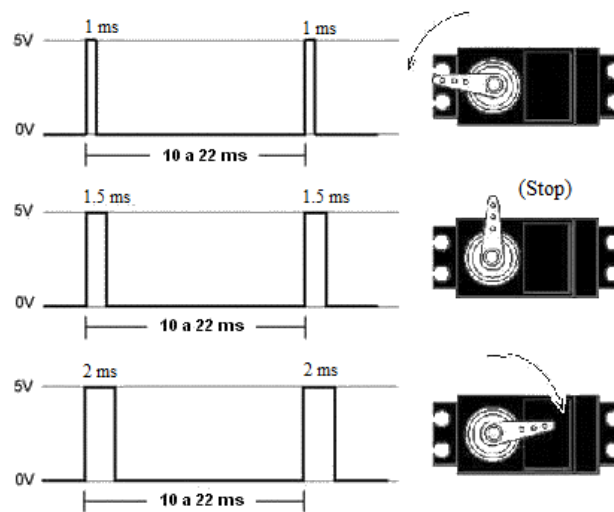
Servo Parallax de rotació continua.

El servomotor està format bàsicament pels següents components: un motor de corrent continua, un joc d'engrenatges reductors que dona força al motor però li resta velocitat, un potenciòmetre per la realimentació de la posició i electrònica de control. El senyal d'entrada serà comparada gracies a aquests component el que permetrà, a diferencia dels motors de continua, posicionar-nos, i mantenir la posició, dins el rang de treball.

El servomotor està controlat mitjançant un senyal PWM⁷, senyal de pols modulats, que té un període aproximat de 20 ms i on el pols ha d'estar en alt entre 1 i 2 ms, cosa que determinarà el sentit i velocitat de gir del servomotor. El senyal d'entrada està estandarditzada per facilitar el control d'aquests tipus de servos. El senyal de

⁷ PWM (en anglès: Pulse With Modulation) és una tècnica que es basa en la variació del pols, o part positiva, d'una senyal periòdica, en relació al seu període.

referència de parada o centre, és un pols d'ona quadrada de 1.5 milisegons amb un període entre 10 i 22 ms. Mentre el pols es mantingui en aquest ample, el servomotor es situarà a la posició central i es mantindrà parat. Si l'ample de pols es disminueix o augmenta entre uns valors de 1 ms i 2 ms, el servo es mourà de manera proporcional cap a un costat o cap a l'altre. Un servo normal té un rang entre 90 i 180 graus de circumferència, limitant el seu moviment a aquest rang. El nostre servo al ser de rotació continua, girarà indefinidament fins que canviï l'ample del pols.



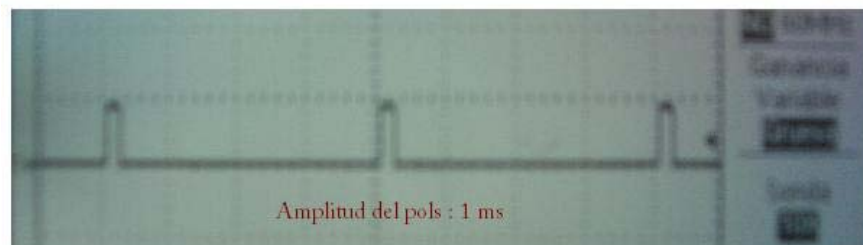
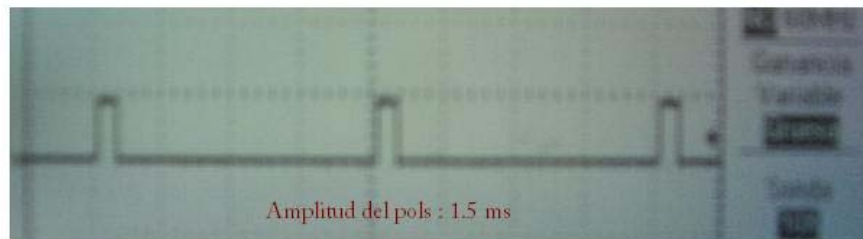
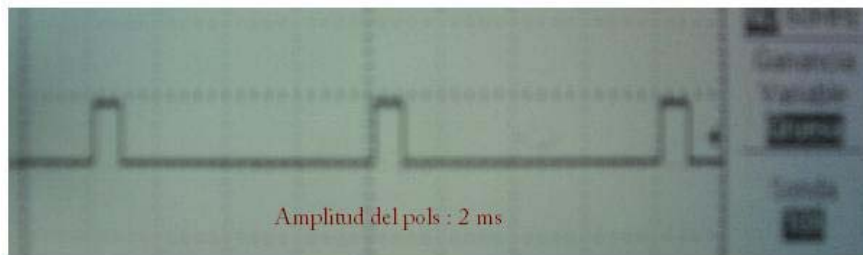
Relació entre la senyal d'entrada i el moviment del servomotor

Per a la generació de l'ona PWM amb el microcontrolador, el més habitual és fer ús d'un timer i un comparador, i les seves interrupcions associades. D'aquesta manera el microcontrolador queda lliure per poder realitzar altres tasques, i la generació del senyal és automàtica i més efectiva.

El mecanisme consisteix en la programació del timer perquè generi per si sòl una ona quadrada amb l'ample que volem. Hem de configurar el timer perquè treballi en mode Fast PWM, mode que ens permetrà formar l'ona amb només una interrupció. També hem de carregar el valor del registre comparador que determinarà l'ample del pols i un event per determinar el període del pols.

Un cop a començat l'execució, es produirà una interrupció que indicarà que hem arribat al final del període (20 ms). En aquest moment hem de posar les sortides a 1 per generar el pols i finalitzem la interrupció. Segueix l'execució i arribem al moment que el valor del registre comparador (1.5ms, parat) és menor que el del temporitzador.

Gracies al mode Fast PWM, automàticament la línia canvia d'estat, de 1 a 0, fins que tornem a arribar al final del període de l'ona.



Generació d'ones de període 20ms i amples de pols de 2 ms, 1.5ms, 1 ms

Utilitzant aquesta característica només necessitem una interrupció cada 20 ms.

Fins ara hem parlat sobre tot de l'aspecte de disseny físic o hardware del projecte. Ara parlarem una mica sobre el software necessari que hem escollit per fer funcionar aquest projecte.

AVR Studio 4 - WinAVR

AVR Studio és un software de programació de microcontroladors de la casa Atmel. La companyia facilita de manera totalment gratuïta aquest software per tal de poder programar els seus microcontroladors. Això ens convé molt si volem reduir el màxim possible els costos.



AVR Studio, software de programació de μ C AVR

El AVR Studio ens permet la programació a baix nivell en llenguatge ensamblador, així com també en llenguatge dalt nivell C, un cop instal·lat el WinAVR (Compilador Gcc per la plataforma AVR). WinAVR és un conjunt d'eines entre les quals s'inclouen: avr-gcc (the command line compiler), avr-libc (the compiler library), avr-as (the assembler), avrdude (the programming interface), avarice (JTAG ICE interface), avr-gdb (the de-bugger), i d'altres. El AVR Studio reconeix e incorpora la instal·lació del WinAVR permetent treballar a partir d'aquest moment en C.

Amb aquest software podrem editar, compilar i “debugar” el codi del nostre programa per tal de poder programar el microcontrolador. També ens permet fer un “debugar” “in-circuit”, el que vol dir que al mateix temps que estem executant el programa al microcontrolador, podem fer una revisió línia a línia del nostre programa, veient l'estat dels diferents registres, i recursos del μ C.

El AVR Studio es una eina molt potent, eficaç i té la característica que el seu ús es molt senzill, podent començar amb la programació des del primer dia.

Drivers de control dels sensors

També s'inclouen amb aquest projecte, els drivers d'ús dels sensors utilitzats pel Robot. Per tal que l'usuari pugui començar a utilitzar el robot de manera simple i sense massa coneixement intern del robot, s'han creat uns drivers de control per a cada un dels dispositius sensors.

Es poden veure els drivers en més detall, així com el mode d'ús en l'apèndix d'aquest document.

RESULTATS I CONCLUSIONS

Un cop realitzat el projecte i dissenyat i construït el model de robot, és hora d'avaluar els resultats obtinguts en cada decisió presa al llarg del desenvolupament del projecte.

En un primer moment es va considerar l'alternativa de l'Scribbler, però es va desestimar ràpidament per tenir moltes limitacions i no complir diversos requeriments de disseny. Es va optar per la construcció d'un petit prototip per recrear el que buscàvem en el model anterior però mirant d'incloure més funcionalitats.

Al llarg del desenvolupament d'aquest projecte s'ha vist que la AVR atmega128 és una millor alternativa per les seves capacitats d'e/s, velocitat de procés i recursos disponibles. Amb el nou robot aconseguim tenir molt més capacitat de càlcul, cosa que anirà molt bé si volem afegir nous component o volem expandir el model actual. La capacitat de connexió i d'expansió s'ha vist ampliada enormement, permetent ampliar en gran manera el robot actual. En el referent als costos de disseny, hem aconseguit crear un petit prototip reduint el màxim els costos per tal de que sigui un model viable.

L'objectiu principal d'aquest projecte ha estat crear una eina d'aprenentatge de diversos aspectes de l'enginyeria, sense que sigui massa complicat per aquells alumnes que comencen a estudiar el tema. Això penso que s'ha aconseguit tractant d'utilitzar una arquitectura el més entenedora possible, unes eines de treball potents però fàcils en el seu ús, i una varietat de funcionalitats que permetessin englobar els màxims aspectes possibles d'una enginyeria.

Milliores i Possibles Ampliacions

Com totes les coses, res és perfecte, aquest projecte es pot millorar en alguns aspectes, i també s'hi poden implementar noves característiques per aconseguir una experiència educativa més completa.

Aquest model és un prototip, i el muntatge final, està encara una mica desprotegit. Es podria dissenyar una coberta de recobriment o una estructura compacta o anessin els

sensors i que cobris l'interior del robot. D'aquesta manera aconseguiríem donar més consistència al robot i dotar-lo de protecció enfront de caigudes o cops.

Una altre millora podria ser afegir nous sensors per fer més completa la percepció de l'entorn per part del robot. Aquesta millora implicaria l'estudi de nous sensors, és a dir noves maneres de detecció i interacció amb l'entorn, cosa que seria molt interessant.

Una possible ampliació seria estudiar la manera d'afegir al microcontrolador un sistema operatiu en temps real com pot ésser per exemple el FreeRTOS. D'aquesta manera se'ns permetria l'ús de processos i el S.O s'encarregaria de gestionar aquests processos i repartir l'ús de recursos. El FreeRTOS és un projecte de codi lliure que està escrit en codi C, cosa que afavoriria la unió amb el projecte actual.

Afegir un sistema operatiu com aquest, encara que és un sistema operatiu molt bàsic, permetria a la vegada l'estudi de molts aspectes relacionats amb els sistemes operatius, obrint encara més les matèries d'aplicació del robot.

Per últim un altre ampliació, més complicada l'anterior, seria la integració del robot en l'entorn Pyro. Pyro és un entorn de programació i simulació de robots, ja poden ser reals o virtuals, simulant el seu funcionament i el seu comportament en diferents circumstàncies. Està escrit totalment en python, un llenguatge interpretat d'alt nivell molt potent que es podria utilitzar per la programació.

Amb la configuració actual del robot ser impossible la incorporació de l'entorn pyro així com del llenguatge python. Seria necessari la incorporació d'un petit computador o semblant que pogués suportar aquest entorn. Aquesta idea seria molt interessant ja que l'entorn pyro, obre un ventall de possibilitats per l'estudi de la programació de comportaments del robot.

Llibres :

1. Gadre, Dhananjay V. *Programming and customizing the AVR microcontroller*. Nueva York: McGraw-Hill, 2001. 339p.
2. Parab, J.S., & Shelake, V. G., & Kamat, R. K., & Naik, G. M., *Exploring C for Microcontrollers*. Dordrecht: Springer, 2007. 157p.
3. Morton, John. *AVR An Introductory Course*. Oxford: Newnest, 2002. 241p.

Adreces electròniques :

4. Atmel Corporation. *ATMEGA128 (L) Summary*. [En línia]. 2002. Atmel Corporation, 2002, última revisió : 2007. [2007-2008] http://www.atmel.com/dyn/resources/prod_documents/2467S.pdf
5. Atmel Corporation. *ATMEGA128 (L) Reference*. [En línia]. 2002. Atmel Corporation, 2002, última revisió : 2007. [2007-2008] http://www.atmel.com/dyn/resources/prod_documents/doc2467.pdf
6. Peter Fleury. *AVR-GCC Software* [En línia]. 2006. Peter Fleury., 2006-2007, [2007-2008].
<http://homepage.hispeed.ch/peterfleury/avr-software.html>
7. AVR Freaks. *AVR. The nr.1 AVR Comunity* [En línia]. 2004. AVR Freaks, 2004-2008, [2007-2008]
<http://www.avrfreaks.net/>
8. Microchip Technology Inc. *PIC18F8722 Family Technical data* [En línia]. 2004 . Microchip Technology Inc., 2004, última revisió: 2004 [2007-2008]

<http://ww1.microchip.com/downloads/en/DeviceDoc/39646b.pdf>

9. Freescale semiconductor. MC68HC11F1 *Technical data* [En línia]. 2004 .
Freescale semiconductor, 2004, última revisió: 2004 [2007-2008]
http://www.freescale.com/files/microcontrollers/doc/data_sheet/MC68HC11F1.pdf

APÈNDIX I

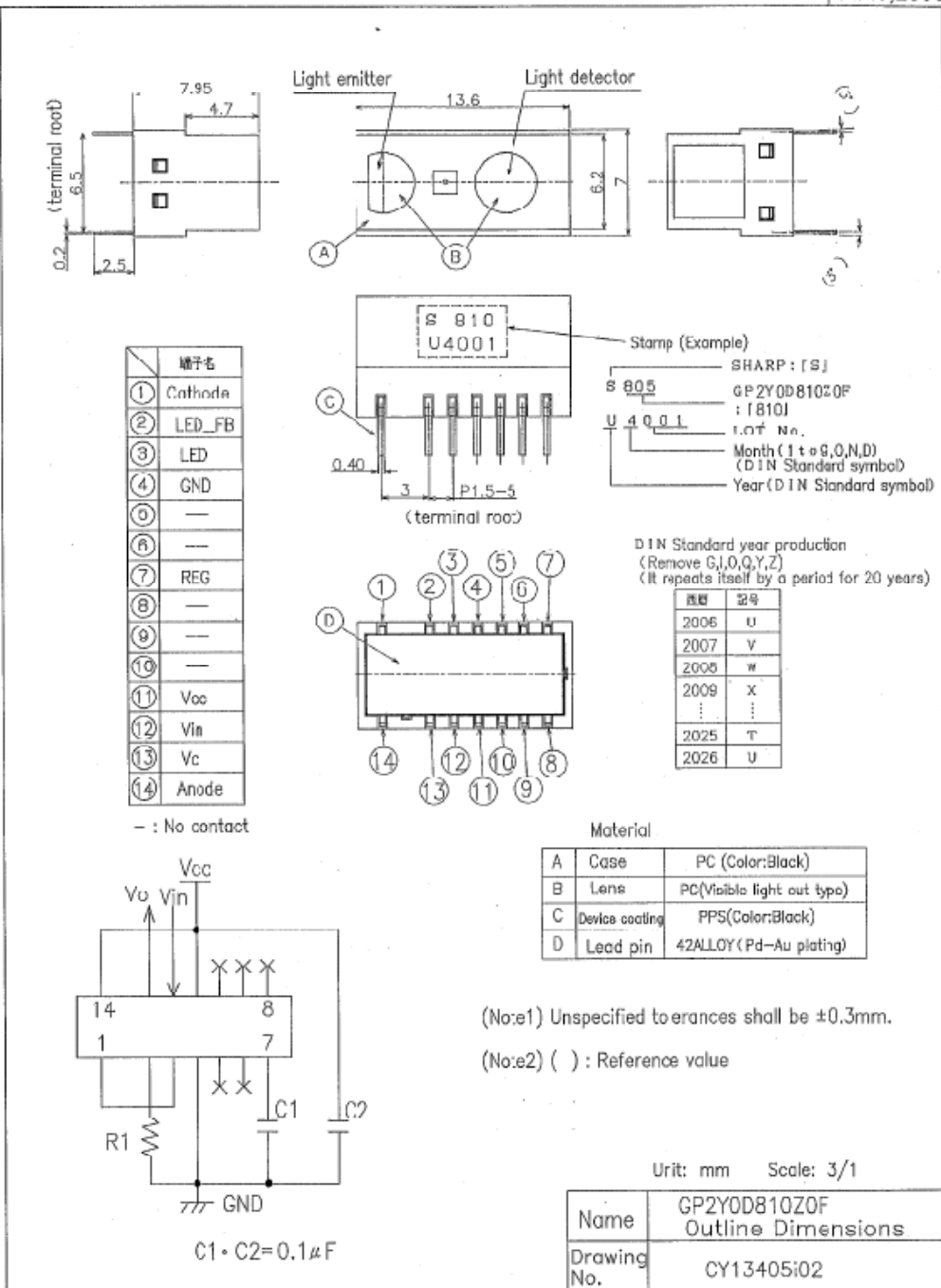
Especificacions tècniques

Sensor Infraroig - SHArp GP2Y0D805Z0F

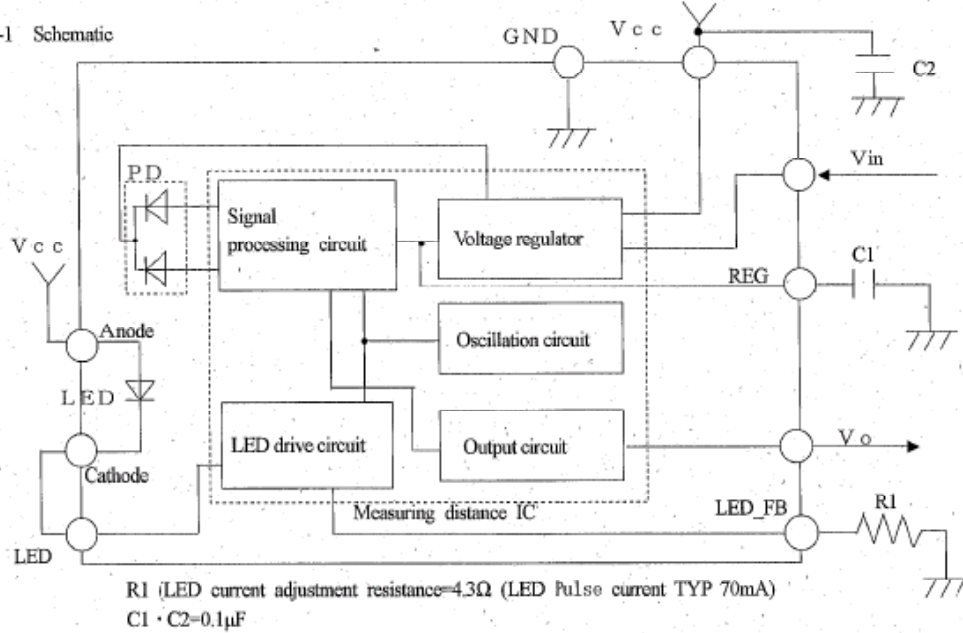
Font d'informació: Document tècnic oficial de Sharp.

2. Outline

ED-06G025A GP2Y0D810Z0F
April.20,2006



3-1 Schematic



3-2 Absolute maximum ratings

Parameter	Symbol	Ratings	Unit	Remark
Supply voltage	Vcc	-0.3 to +7	V	-
Output terminal voltage	V _o	-0.3 to Vcc+0.3	V	-
Input terminal voltage	V _{in}	-0.3 to Vcc+0.3	V	-
Operating temperature	T _{opr}	-10 to +60	°C	-
Storage temperature	T _{stg}	-20 to +70	°C	-
Soldering temperature	T _{sol}	260	°C	5s or less/time up 2 times t=1.0mm One side board mounting

Recommended operating conditions

Parameter	Symbol	Rating	Unit	Remark
Supply voltage	Vcc	2.7 to 6.2	V	-
High level input voltage	V _{inH}	MIN Vcc-0.2	V	CMOS level signal. Operating
Low level input voltage	V _{inL}	MAX 0.2	V	CMOS level signal. Standby state

3-3 Electro-optical Characteristics

(Ta=25°C, Vcc=5V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Output terminal voltage	VoH	Output voltage at high level	Vcc-0.6	-	-	V
	VoL	Output voltage at low level	-	-	0.6	V
Output distance characteristics	Vo	(*1) (*2)	80	100	130	mm
Average supply current	Icc 1	Vcc=5V Vin=5V R1=4.3Ω (*3)	-	5	6.5	mA
Average supply current	Icc 2	Vcc=5V Vin=5V R1=4.3Ω (*3)	-	9	10.5	mA
Stand-by supply current	Icc 3	Vcc=5V Vin=0V	-	5	8	μA

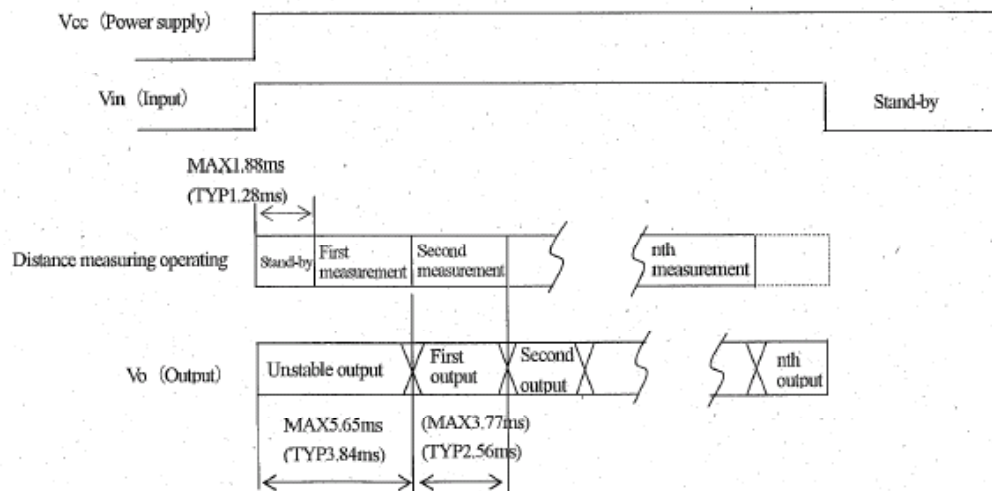
※ L: Distance to reflective object

(*1) Using reflective object: White paper (Made by Kodak Co., Ltd. gray cards R-27•white face, reflectance; 90%)

(*2) Output switching has a hysteresis width.
The distance specified by Vo should be the one with which the output H switches to the output L.

(*3) Icc1: (LED Emitting time: TYP20 μs*8times) Icc2: (Emitting time: TYP20 μs*15times)
LED Pulse Current: TYP 70mA

3-4 Timing chart



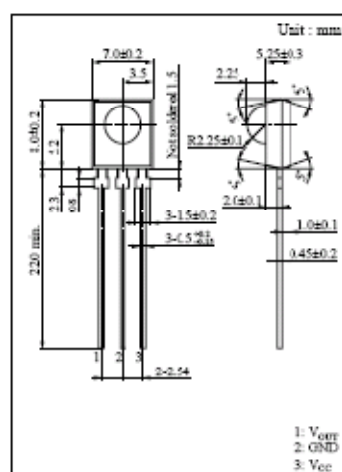
Panasonic

Bipolar Integrated Circuit with Photodetection Function

For infrared remote control systems

- Extension distance is 8 m or more
- External parts not required
- Adoption of visible light cutoff resin

Parameter	Symbol	Ratings	Unit
Power supply voltage	V_{CC}	-0.5 to +7	V
Power dissipation	P_D	200	mW
Operating ambient temperature	T_{opr}	-20 to +75	°C
Storage temperature	T_{stg}	-40 to +100	°C



Parameter	Symbol	Conditions	min	typ	max	Unit
Operating supply voltage	V_{CC}		4.7	5.0	5.3	V
Current consumption	I_{CC}	Note 3	1.8	2.4	3.0	mA
Maximum reception distance	L_{max}	Note 1	8	10		m
Low-level output voltage	V_{OL}	Note 2		0.35	0.5	V
High-level output voltage	V_{OH}	Note 3	4.8	5.0	V_{CC}	V
Low-level pulse width	T_{WL}	Note 1	200	400	600	μs
High-level pulse width	T_{WH}	Note 1	200	400	600	μs
Carrier frequency	PNA1601M	f_0		36.7		kHz
	PNA4602M			38.0		
	PNA4603M			36.9		
	PNA4610M			33.3		

Note 3) Light shut off condition

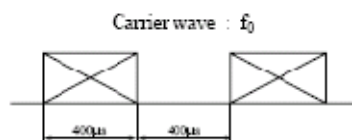


Fig. 1

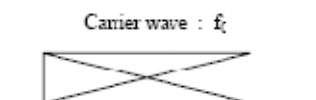
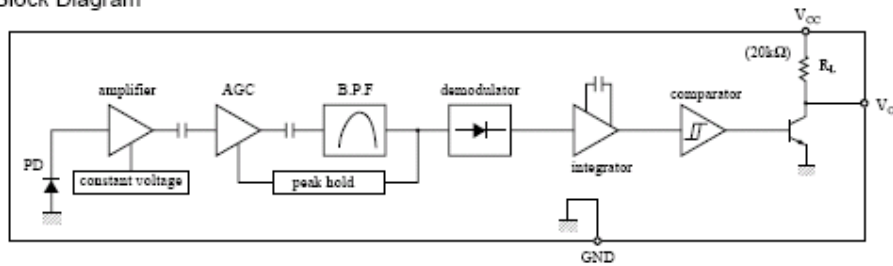
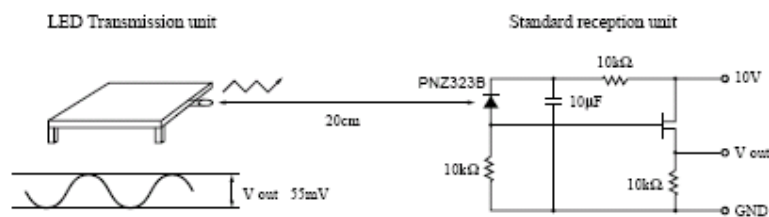


Fig.2

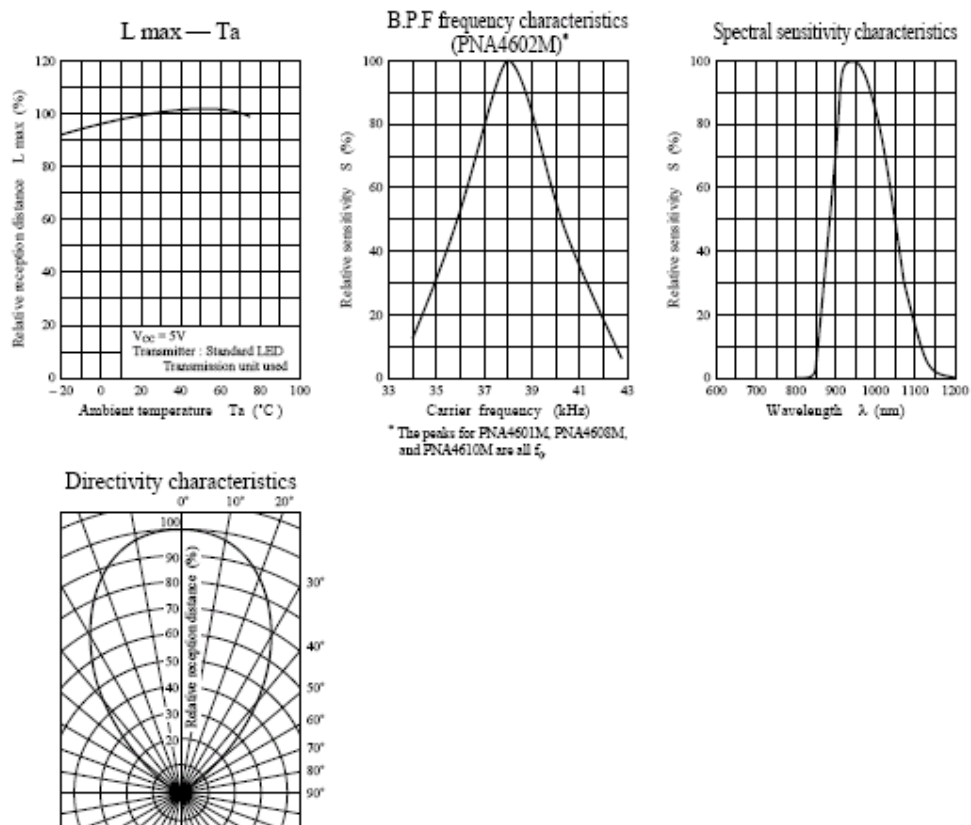
■ Block Diagram



■ Panasonic Transmitter Specifications



- The light output of the LED transmission unit is adjusted so that the transmission output (V_{out}) of the standard reception unit will be 55 mV when the transmission waveform (duty = 50%) is output from the LED transmission unit. Here, infrared sensitivity (SIR) of PNZ323B is 0.53 μA when emission illuminance (H) is 12.45 $\mu\text{W}/\text{cm}^2$.
- The maximum reception distance under these specifications is an assurance that T_{WH} and T_{WL} values will be within the tolerance ranges when 16 consecutive pulses of an optical output equivalent to the maximum reception distance are transmitted by the above transmission unit (The maximum reception distance is measured in the dark without external disturbance noise.)



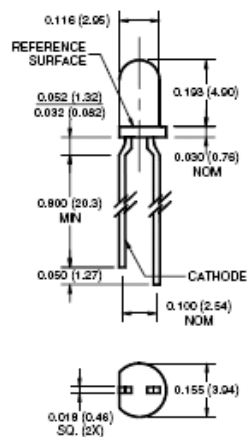


PLASTIC INFRARED LIGHT EMITTING DIODE

QEC112

QEC113

PACKAGE DIMENSIONS

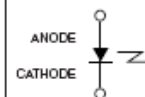


NOTES:

1. Dimensions for all drawings are in inches (mm).
2. Tolerance of $\pm .010$ (.25) on all non-nominal dimensions unless otherwise specified.



SCHEMATIC



DESCRIPTION

The QEC11X is an 940 nm GaAs LED encapsulated in a clear peach tinted, plastic T-1 package.

FEATURES

- $\lambda = 940$ nm
- Chip material = GaAs
- Package type: T-1 (3mm)
- Matched Photosensor: QSC112
- Narrow Emission Angle, 24°
- High Output Power
- Package material and color: Clear, peach tinted plastic

TYPICAL PERFORMANCE CURVES

Fig.1 Normalized Radiant Intensity vs. Forward Current

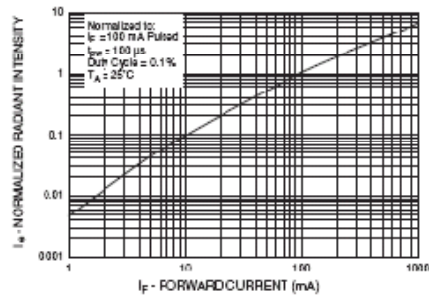


Fig.2 Coupling Characteristics of QEC11X And QSC11X

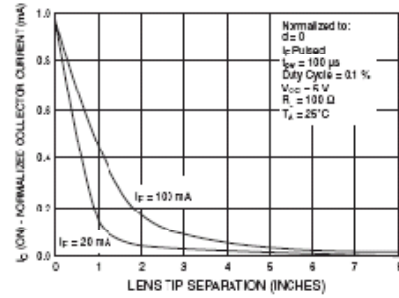


Fig.3 Forward Voltage vs. Ambient Temperature

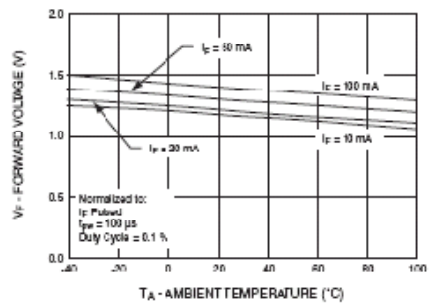


Fig. 4 Normalized Intensity vs. Wavelength

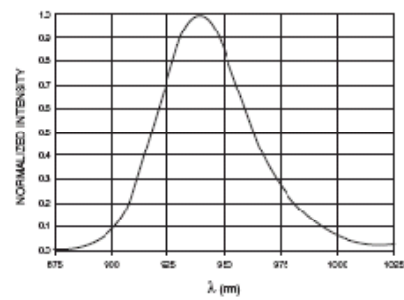
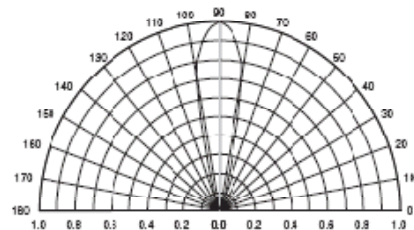
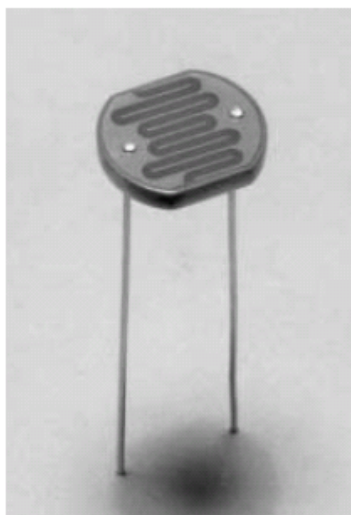
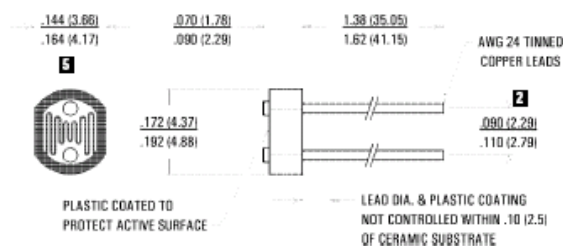


Fig.5 Radiation Diagram




VT900 Series

**PACKAGE DIMENSIONS** inch (mm)

ABSOLUTE MAXIMUM RATINGS

Parameter	Symbol	Rating	Units
Continuous Power Dissipation Derate Above 25°C	P_D $\Delta P_D / \Delta T$	80 1.6	mW mW/°C
Temperature Range Operating and Storage	T_A	-40 to +75	°C

ELECTRO-OPTICAL CHARACTERISTICS @ 25°C (16 hrs. light adapt, min.) **4**

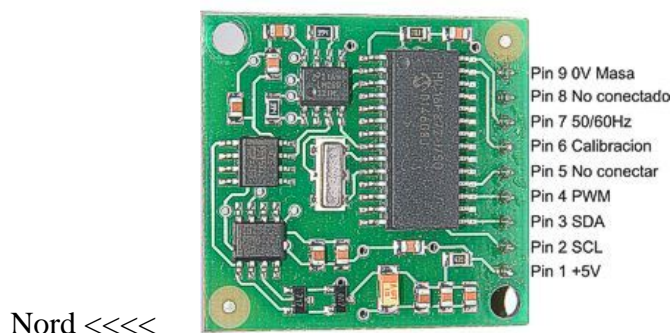
Part Number	Resistance (Ohms) 						Material Type	Sensitivity (γ , typ.) <small>100 (R100R10) 1000 (R100R1)</small>	Maximum Voltage (V, pk)	Response Time @ 1 f _c (ms, typ.)	
	10 lux 2850 K			2 fc 2850 K	Dark					Rise (1-1/e)	Fall (1/e)
	Min.	Typ.	Max.	Typ.	Min.	Sec.					
VT90N1	6 k	12 k	18 k	6 k	200 k	5	Ø	0.80	100	78	8
VT90N2	12 k	24 k	36 k	12 k	500 k	5	Ø	0.80	100	78	8
VT90N3	25 k	50 k	75 k	25 k	1 M	5	Ø	0.85	100	78	8
VT90N4	50 k	100 k	150 k	50 k	2 M	5	Ø	0.90	100	78	8
VT93N1	12 k	24 k	36 k	12 k	300 k	5	3	0.90	100	35	5
VT93N2	24 k	48 k	72 k	24 k	500 k	5	3	0.90	100	35	5
VT93N3	50 k	100 k	150 k	50 k	500 k	5	3	0.90	100	35	5
VT93N4	100 k	200 k	300 k	100 k	500 k	5	3	0.90	100	35	5
VT935G											
Group A	10 k	18.5 k	27 k	9.3 k	1 M	5	3	0.90	100	35	5
Group B	20 k	29 k	38 k	15 k	1 M	5	3	0.90	100	35	5
Group C	31 k	40.5 k	50 k	20 k	1 M	5	3	0.90	100	35	5

See page 13 for notes.

La brúixola digital CMPS03 és un sensor de camps magnètics que una cop calibrat ofereix una precisió de 3-4 graus i una resolució de dècimes. Té dos interfícies, mitjançant polsos temporitzats (modulació en amplada), o bé, mitjançant un bus I2C, que facilita la seva comunicació amb una extensa gamma de microcontroladors, incloent els Basic Stamp, Basic X, OOPIC i altres llenguatges compilats. Aquest sensor magnètic esta especialment dissenyat com a sistema de navegació per a robots. La brúixola esta basada en els sensors KMZ51 de Phillips que són suficientment sensibles com per captar el camp magnètic de la terra. Utilitzant dos d'aquests sensors col·locats en un angle de 90 graus, permet al microprocessador calcular la direcció del component horitzontal del camp magnètic natural.

Connexionat i funcionament

A la següent imatge es poden observar les connexions dels diferents pins del CMPS03, entre els que destaca el pin 4 i el 2-3, que es corresponen als dos possibles interfícies que incorpora: el PWM (Pulse Width Modulation – Modulació per amplada de pols) i el bus I2C format pels senyals SDA (senyal de dades) i SCL (senyal de rellotge) i que és un interfície sèrie bidireccional sincronitzat.

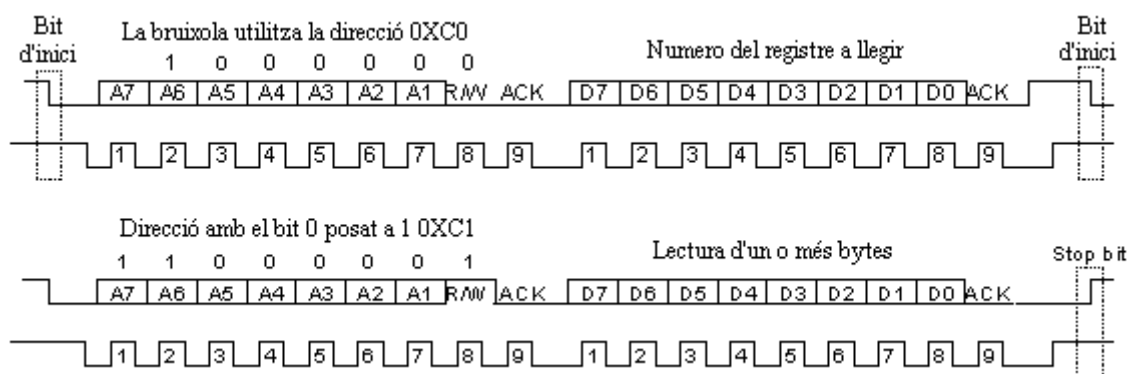


A la sortida 4, s'obté una senyal PWM a la que el pols positiu representa l'angle de la brúixola. El pols varia en duració des de 1mS (0°) fins a 36,99 ms (359,9°), o dit d'una altra manera, el pols és igual a 100 us per cada grau més 1ms de tara. El senyal es manté a zero durant 65 ms entre polsos, per lo que el període de treball és de 65 ms +

l'amplada del pols. El pols és generat per un controlador de 16 bits del propi processador, amb una resolució de 1 uS, encara que en la practica no és recomanable fer mesures amb una resolució de més de 0'1° (10uS). Quan s'utilitzi una interfície PWM, és necessari connectar a +5V mitjançant 2 resistències de 47 Kohm, els pins 2 i 3 (SCL – SDA) de la interfície I2C, ja que no s'inclouen resistències de pull-up al circuit.



L'altra possibilitat és la d'utilitzar la interfície I2C format pels pins 2 i 3, que ens permet una lectura directa del valor en graus de la direcció. El protocol de comunicació I2C utilitzat en el mòdul, és el mateix que s'utilitza amb les populars eeprom com la 24CO4. Primer s'envia un bit d'inici, la direcció del mòdul (0XC0) amb el bit de lectura a zero, i a continuació el número de registre que es desitja llegir. Després es torna a enviar el bit d'inici i la direcció del mòdul a 1 (0XC1). Ara es pot llegir un, o els dos bytes corresponents als registres de 8 i 16 bits respectivament. (El byte de major pes es llegeix primer als registres de 16 bits).



Registres

La brúixola té un total de 16 bytes de registres, alguns dels quals formen registres de 2 bytes tal i com pot observar-se a la següent taula:

Registre	Funció
0	Numero de Revisió del Software
1	Direcció en 1 byte 0-255 para 0 - 360°
2,3	Dirección en 2 bytes 0-3599 para 0 - 359,9°
4,5	Test intern senyal diferencial sensor 1
6,7	Test intern senyal diferencial sensor 2
8,9	Test intern, valor de calibratge 1
10,11	Test intern, valor de calibratge 2
12	Sense ús, torna 0
13	Sense ús, torna 0
14	Sense ús, torna 0
15	Comanda de calibratge, escriure 255 per calibrar

El registre 0 és la revisió del software que actualment és el 8. El registres 1 és la direcció, en graus, convertida a un valor entre 0 i 225 i que pot ser molt útil en certes aplicacions a on resulta complicat utilitzar l'escala de 0 a 360 graus que requereix dos bytes i que esta disponible als registres 2 i 3 (el 2 és més significatiu) amb un valor que va entre 0 i 3599 que equival a 0 -359,9 graus. Els registres 4 a 11 són d'utilització interna i del 12 al 14 no s'utilitzen, pel que no és necessari llegir-los amb la finalitat de no consumir l'ampli de banda del bus I2C. El registre 15 s'utilitza per calibrar la brúixola tal i com s'especifica més endavant.

El bus I2C del circuit no incorpora les resistències necessàries de pull-up, per aquesta raó serà necessària la seva implementació en el mateix, per aquesta raó és recomanable utilitzar dues resistències de 1K8 en cas d'utilitzar el bus a 400 KHz i de 1K2 o 1K si s'utilitza a una freqüència d'1Mhz. Només són necessàries 2 resistències en total per a tot el bus, no per a cada circuit que està connectat al mateix. El sensor de brúixola digital esta dissenyat per a ser compatible amb la velocitat estàndard de rellotge de 100 KHz, encara que aquesta pot augmentar-se si es té en compte el següent:

A velocitats superiors als 160 KHz, la CPU no pot respondre de manera suficientment com per a llegir les dades, per aquesta raó cal incorporar un retard de 50 us al finalitzar l'escriptura del registre de direcció. Si es fa de forma correcta, és possible comunicar amb el mòdul a velocitats superiors a 1 MHz. Això només afecta a programes escrits en llenguatges d'alta velocitat i baix nivell, com és l'assemblador, i no afecta a les aplicacions escrites pels compiladors interns com són el Basic Stamp, l'OOPIC o el Basic X o similars. El mòdul de sensor de brúixola sempre actua com un esclau, mai com un màster del bus I2C.

El pin 7 s'utilitza per a seleccionar entre 50 Hz (posat a 0) o 60 Hz (posada a 1). Això és degut a una desviació errònia d'uns 1,5° causada pel camp generat per la xarxa elèctrica. Sincronitzant la conversió amb la freqüència d'hertz de la xarxa, s'aconsegueix disminuir l'error a tan sols 0,2°. El pin si té una resistència interna de pull-up, per la qual cosa si que es deixa sense connectar, funcionarà a 60 Hz. El circuit realitza una conversió interna cada 40ms (50 Hz) o cada 33,3ms (60Hz) d'acord amb la connexió d'aquesta entrada. No hi ha cap tipus de sincronisme entre la realització de la conversió i la sortida de dades, ja que quan aquests son llegits es retorna el valor més recent que estigui guardat al seu respectiu registre.

El pin 6 s'utilitza per calibrar el sensor magnètic. Aquesta entrada té la seva pròpia resistència de polarització (pull up) i pot deixar-se sense connectar una vegada realitzada la conversió.

Els pins 5 i 8 estan marcats com no connectats, encara que el pin 8 és en realitat el reset del microprocessador, amb la finalitat de poder programar-lo una vegada soldat al circuit imprès. Aquesta entrada no té resistència pull-up.

Calibratge

ATENCIÓ: Abans de realitzar la calibratge, el mòdul deurà mantenir-se perfectament horitzontal amb els components cap amunt i els dos sensors a la cara inferior. Mantenir el mòdul allunyat d'objectes metàl·lics i molt especialment d'objectes magnètics com imants i altaveus. També és necessari conèixer amb precisió la direcció en la que es troben els quatre punts cardinals, per aquesta raó és **absolutament necessari** comprovar-lo amb una brúixola magnètica.

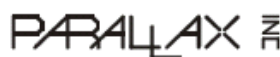
El calibratge de la brúixola digital pot fer-se mitjançant qualsevol dels següents mètodes:

1. El Mètode I2C → Aquest mètode consisteix a escriure 225 al registre 15 del mòdul per cada un dels 4 punts cardinals. El valor 255 s'esborra internament cada vegada que es completa el calibratge. Els punts de calibratge poden fer-se en qualsevol ordre, però sempre és necessari calibrar els 4 punts. Per exemple:

1. Apunta el circuit cap al Nord. Escriu 255 al registre 15
2. Apunta el circuit cap a l'Est. Escriu 225 al registre 15
3. Apunta el circuit cap al Sud. Escriu 225 al registre 15
4. Apunta el circuit cap a l'Oest. Escriu 225 al registre 15

2. El Mètode del polsador → Consisteix a utilitzar el polsador entre massa i el pin 6 del circuit, amb la finalitat d'iniciar el calibratge. S'ha de tenir en compte que aquest pin té una resistència de polarització interna i es pot deixar sense connectar una vegada feta el calibratge. Per a realitzar el calibratge, només cal posar a massa el pin 6 momentàniament per cadascun dels punts cardinals. De la mateixa manera que amb l'altre mètode, els punts poden calibrar-se en qualsevol ordre, però sempre és necessari calibrar els 4 punts cardinals. Exemple:

1. Apunta el circuit cap al Nord. Polsa momentàniament en polsador.
2. Apunta el circuit cap a l'Est. Polsa momentàniament en polsador.
3. Apunta el circuit cap a l'Oest. Polsa momentàniament en polsador.
4. Apunta el circuit cap al Sud. Polsa momentàniament en polsador.



599 Main Drive, Suite 100
Rocklin, California 95765, USA
Office: (916) 624-8333
Fax: (916) 624-8003

General: info@parallax.com
Technical: support@parallax.com
Web Site: www.parallax.com
Educational: http://www.parallax.com/html_pages/edu/

Continuous Rotation Servo (#900-00008)

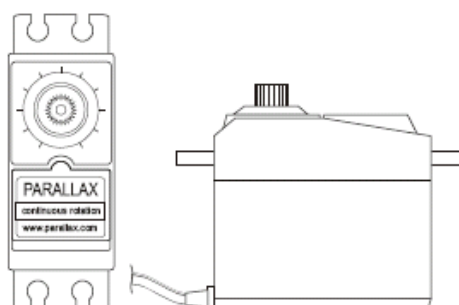
General Information

The Parallax Continuous Rotation servo is ideal for robotic products that need a geared wheel drive or other projects that require a 360 degree rotation geared motor. The Parallax Continuous Rotation servo output gear shaft is a standard Futaba configuration. The servo can be adjusted with a small Phillips screw driver if the unit becomes out adjustment on its center set point. Servo is custom manufactured for Parallax by Futaba.

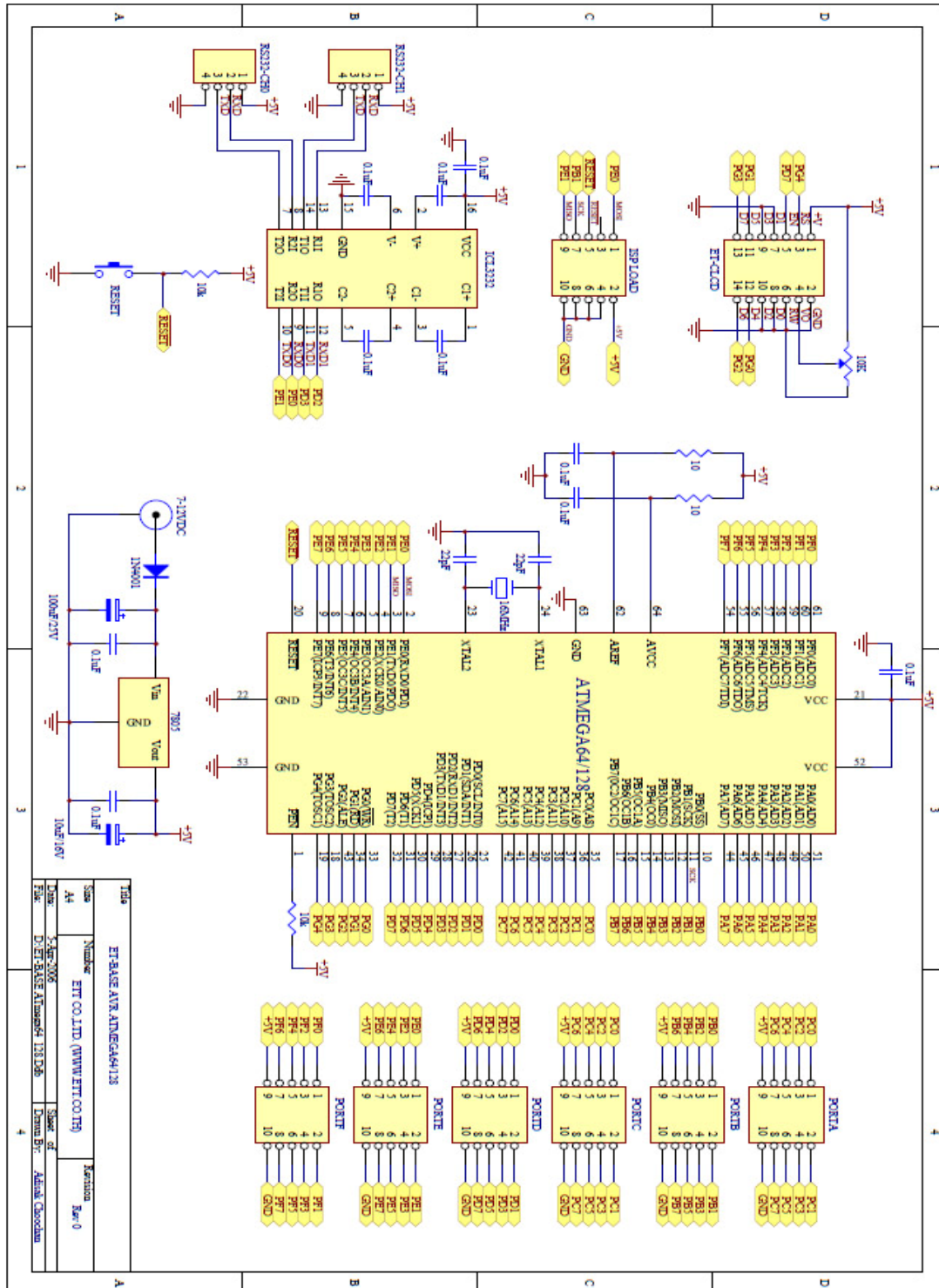


Technical Specifications

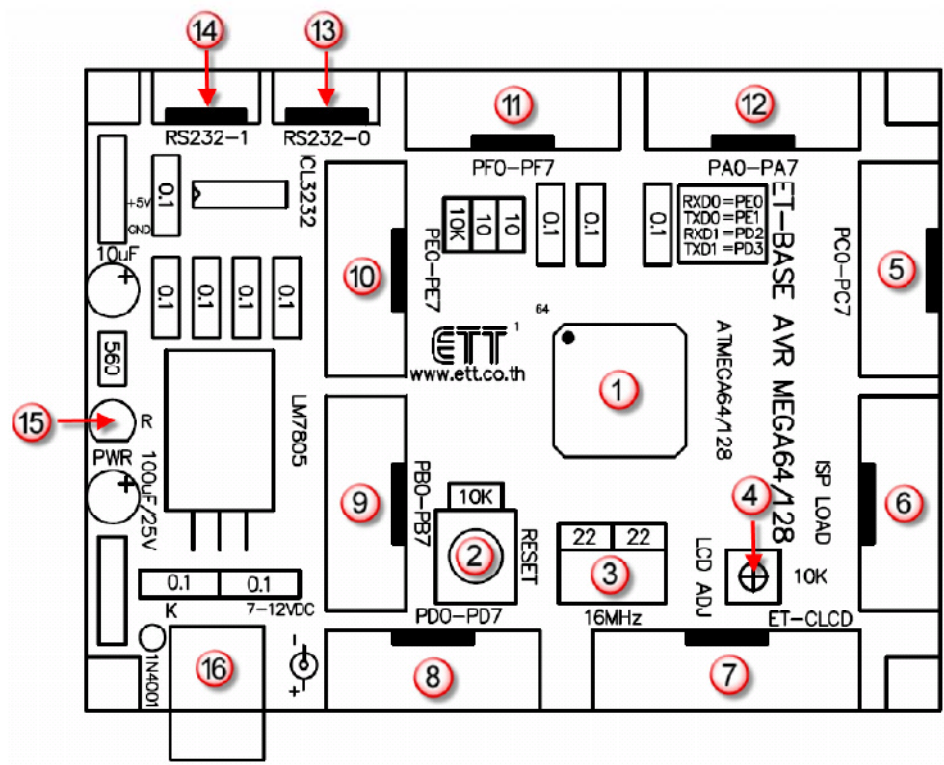
- > Power 6vdc max
- > Average Speed 60 rpm
Note: with 5vdc and no torque
- > Weight 45.0 grams/1.59oz
- > Torque 3.40 kg-cm/47oz-in
- > Size mm (L x W x H)
40.5x20.0x38.0
- > Size in (L x W x H)
1.60x.79x1.50
- > Manual adjustment port



Esquema:



Estructura de la placa:



1. Microcontrolador Atmel AVR ATMEGA 128.
2. Botó de reset
3. Cristall oscil·lador de 16Mhz.
4. Resistència pel control del contrast del LCD
5. Port C 8-bit; PC0-PC7
6. Port ISP (In circuit Serial Programming).
7. Port ET-CLCD de connexió de pantalla LCD en mode 4 bits.
8. Port D 8-bit; PD0-PD7
9. Port B 8-bit; PB0-PB7
10. Port E 8-bit; PE0-PE7
11. Port F 8-bit; PF0-PF7 (Programador JTAG)
12. Port A 8-bit; PA0-PA7
13. Connector RS232 d'us general.
14. Connector RS232 d'us general.
15. Led d'estat d'alimentació.
16. Connector d'alimentació.

Llista de costos 1ª versió:

Material scribbler (segons sch)					Alternativa AVR	
Component	#	cost/unit.	total	ref.		
Placa						
Basic stamp 2	1	67,980 €	67,980 €	111235	AVR	13,420 €
					Atmega 128	
Alimentació			9,713 €			9,713 €
Regulador volt:						
5V	2	0,320 €	0,640 €	9489436		
Control alimentació	?		#iVALOR	!		
Interruptor	1	2,580 €	2,580 €	1082289		
Fusible 1.1A	1	0,460 €	0,460 €	1175844		
Protafusible	1		0,000 €			
Led	1	0,133 €	0,133 €	1003214		
				753-		
Portapiles	1	5,900 €	5,900 €	00001		
Circuit RS232			5,791 €			3,950 €
Connector DB9	1	2,630 €	2,630 €	1087522		2,630 €
Transistor NPN	2	0,146 €	0,292 €	1017657		
Transistor PNP	2	0,062 €	0,124 €	356554		
Resist 10K	7	0,260 €	1,820 €	1265079		
Resist 4.7K	1	0,260 €	0,260 €	1265077		
Resist 100K	1	0,260 €	0,260 €	1265085		
Condensador 0.1uF	3	0,135 €	0,405 €	1216444		
Pulsador Reset	1		0,000 €			
CDS light sensors			5,319 €			5,319 €
CDS	3	1,330 €	3,990 €	1226438		
Res 220R	3	0,260 €	0,780 €	1265068		
Cap 0.01uF	3	0,183 €	0,549 €	1216435		
Line detector			4,460 €			19,270 €
Fotod. IR receptor	2	2,230 €	4,460 €	1497904		
Fotod. IR emissor	2	0,000 €	0,000 €			
Circuit	?		#iVALOR			
						veure superrobotic a S320115

acondicion.				!	
Motor driver				17,760 €	17,760 €
Amplif. Motor					
bidir.	2		0,000 €		
				900-	
Motor	2	8,750 €	17,500 €	00008	
Circuit					
detector stall	2		0,000 €		
Resist. 1K	1	0,260 €	0,260 €	1265072	
Hacker port				1,379 €	1,379 €
Connector	1	0,200 €	0,200 €		
Leds	3	0,133 €	0,399 €	1003214	
Resistència					
330R	3	0,260 €	0,780 €	1265069	
Altaveu				4,809 €	4,809 €
Altaveu	1	1,430 €	1,430 €	1192971	
LM386M	1	1,170 €	1,170 €	9488324	
Resist 10K	1	0,260 €	0,260 €	1265079	
Resist 2,0K	2	0,260 €	0,520 €		
Resist 680R	1	0,260 €	0,260 €	1265071	
Resist 10R	1	0,260 €	0,260 €	1265059	
Cap 0.1uF	3	0,135 €	0,405 €	1216444	
Cap 10uF	2	0,082 €	0,164 €	9693769	
Cap 220uF	1	0,340 €	0,340 €	9693807	
Detector obstacles				5,830 €	5,830 €
Resist 680R	2	0,260 €	0,520 €	1265071	
Resist 1.0K	1	0,260 €	0,260 €	1265072	
Resist 10.0K	1	0,260 €	0,260 €	1265079	
Emissor IR	2	1,750 €	3,500 €	1045522	
Receptor IR					
integrat	1	1,290 €	1,290 €	4913176	
Mecànica				29,970 €	29,970 €
				700-	
Xasis	1	19,920 €	19,920 €	00022	
Rodes	2	3,350 €	6,700 €	28109	
				700-	
Roda neutra	1	3,350 €	3,350 €	00009	
			#iVALOR		
Cargols i visols	?		!		
		Total	153,011 €		total AVR 110,030 €

La llista de costos anterior, representa totes les peces que està compost el robot Scribbler, per fer una copia idèntica canviant el microcontrolador. Aquesta alternativa no era viable per la complicitat del disseny, i es va optar de muntar un altre robot amb les funcionalitats del Scribbler. En el anterior llista de costos no està contemplat el programador JTAG.

Llista de costos 2^a versió:

Descripció:	Quantitat	Cost / Unit.	Cost
Xassís Boebot	1	16.50 €	16.50 €
Rodes Boebot	2	2.77 €	5.54 €
Roda boja Boebot	1	2.77 €	2.77 €
Servomotor rotació continua Parallax	2	9.11 €	18.22 €
Sensor infraroig PNA4601M	2	2.77 €	5.54 €
Díode emissor infraroig QEC112	2	1.05 €	2.10 €
Infraroig GP2Y0D805Z0F	2	3.49 €	6.98 €
Compàs CMPS03	1	36.34 €	36.34 €
LDR (Fotorresistència) VT900	2	1.39 €	2.78 €
ETT-Base AVR Atmega128	1	22,07 €	22,07 €
Gravador JTAG AVR	1	24,29 €	24,29 €
Placa perforada	1	1.15 €	1.15 €
Components electrònics	1	3 €	3 €
Cables i cargols	1	1 €	1 €
Piles AA	1	1 €	1 €

Total	148.28 €
--------------	-----------------

Aquesta llista de costos correspon a tots els materials utilitzats en el muntatge i programació del robot, però no són els costos reals del projecte. La gran majoria de les peces o components del robot, han estat utilitzades d'altres robots o eren peces disponibles al departament pel seu ús.

Aquest apèndix pretén ésser una petita guia per començar a utilitzar l'aplicació dissenyada. Per tal de poder utilitzar el model que s'ha creat, s'anirà guiant a l'usuari per donar els passos necessaris per poder començar a utilitzar el robot.

Manual D'Usuari

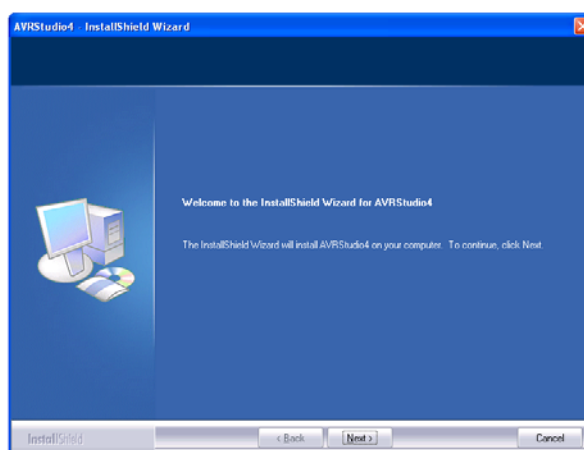
Instal·lació i configuració del Software

Com hem vist en capítols anteriors d'aquest document el software utilitzat consta de bàsicament dos programes: AVR Studio 4 i WinAVR. Aquests dos programes són gratuïts i estan inclosos en el CD del Projecte.

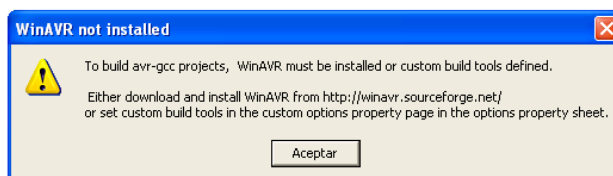
Primer de tot instal·lem el AVR Studio 4 i els dos service pack que s'inclouen en el CD. Acceptem les opcions que ens va oferint l'instal·lador i esperem fins que la instal·lació hagi finalitzat. La instal·lació és molt simple i només s'han de seguir les indicacions.

Un cop instal·lat ja podem començar a utilitzar el programa, encara que només podrem utilitzar llenguatge ensamblador.

Si intentem compilar un programa en C se'ns mostrarà la següent finestra:



Instal·lador AVR Studio



Això vol dir que no pot compilar el llenguatge C i que fa falta instal·lar un compilador per aquest llenguatge. Aquí és on hem d'instal·lar el WinAVR. El WinAVR són un

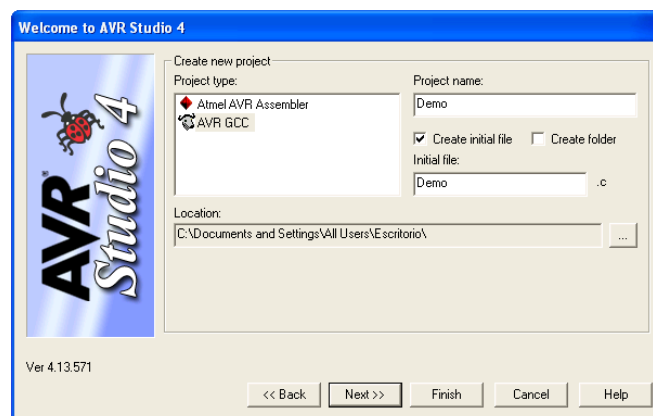
conjunt d'eines per editar i compilar programes en C, té el seu propi editor, però nosaltres seguirem utilitzant l'entorn del AVR Studio. El WinAVR també esta al CD del projecte.

Una cop instal·lats els dos programes ja podem començar a crear el nostre programa.

AVR Studio

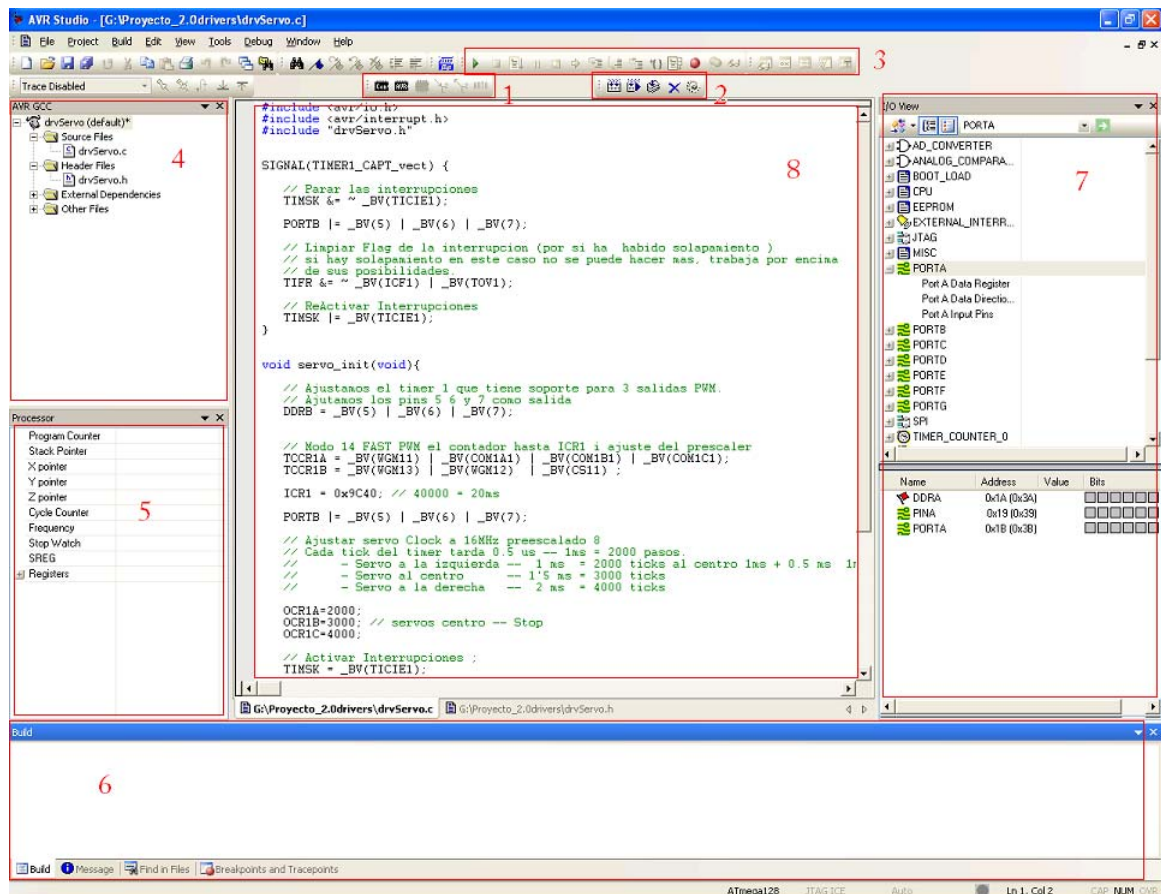
El primer que farem, serà crear un nou projecte per treballar. Al obrir el programa ens apareixerà una finestra per carregar un projecte anterior o crear un de nou si volem.

Crearem un nou projecte per tal de començar a treballar amb el μ C. Seleccionarem el llenguatge que volem utilitzar, assemblador o C, i donarem un nom al projecte. Crearem el projecte donant al botó de finalitzar.



Finestra de creació de nou projecte

L'entorn de programació AVR Studio està format per una sèrie de finestres amb diferents funcions. Podem veure en mes detall el programa a la imatge que es mostra en la pàgina inferior.



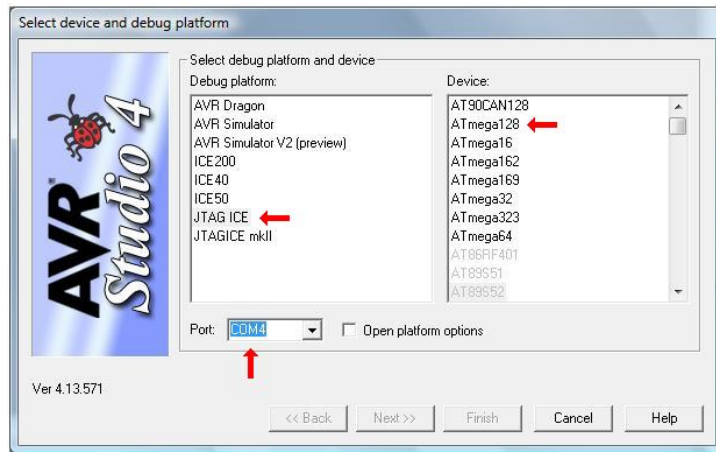
Escriptori de treball AVR Studio 4

1. **Botons de connexió.** Ens permet connectar-nos amb el microcontrolador per descarregar l'arxiu amb programació que hem creat.
2. **Botons de compilació.** Permet compilar el programa i construir els fitxers .hex per carregar en el μC .
3. **Botons de debugació.** Ens permeten debugar el programa pas a pas i repassar tot el codi.
4. **Arxius del projecte.** Finestra on es mostren els arxius que componen el projecte.
5. **Estat del Processador.** Finestra que ens mostra l'estat en cada moment del processador. Aquesta opció només funciona quan estem debugant.
6. **Missatges i errors.** Finestra on se'ns mostren missatges de compilació, debugació i possibles errors del programa.
7. **Estat de l'Entrada/Sortida.** Ens mostra bit a bit l'estat dels registres d'entrada/sortida i d'altres recursos com, conversors, timers memòria,... Com el cas anterior només funciona quan debuguem.
8. **Editor de codi.** Finestra principal on podem escriure i editar el codi de la programació. Aquest codi pot ésser escrit en C o en ensamblador.

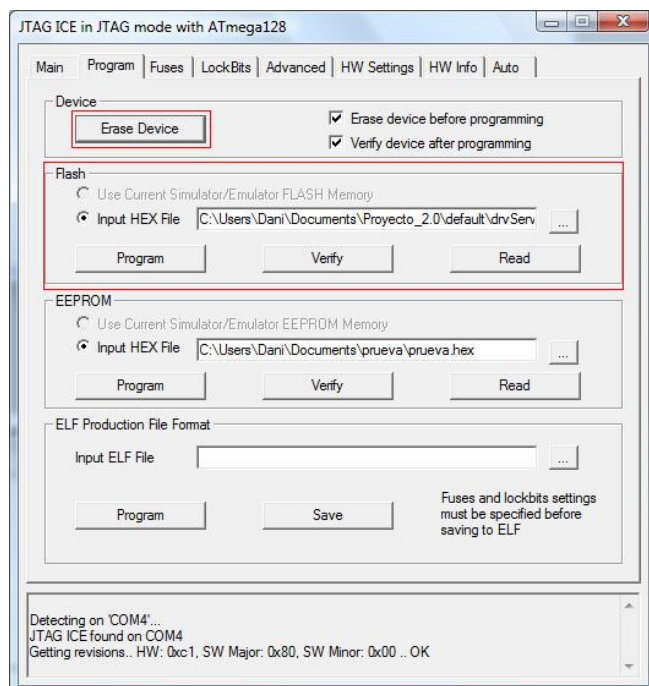
Una vegada creat el nostre programa, és hora de carregar-ho en el microcontrolador i veure el seu funcionament. Primer hem de compilar i crear l'arxiu .hex que carregarem en el microcontrolador. Seleccionem el boto "Compile" i tot seguit "Build". Si hi ha errors debuguem el programa fins que trobem i solucionem aquests errors. Un cop compilat i construït, és hora de carregar-ho al μ C. Utilitzem el botó de "Connect". En aquest punt hem de tenir connectat el PC amb la placa del microcontrolador mitjançant el programador. Nosaltres farem la connexió mitjançant el un programador JTAG.

Seleccionem el tipus de programador que serà mitjançant JTAG, també seleccionem el dispositiu Atmega128 i per últim el port de connexió.

Un cop seleccionat els paràmetres se'ns mostrarà la finestra de la imatge inferior.



Finestra de selecció de programador



En aquesta finestra podem esborrar o gravar el dispositiu. Nosaltres treballem amb la memòria Flash, per tant seleccionarem l'arxiu .hex que hem creat amb el nostre programa i el carregarem amb el botó "Program".

També podem descarregar el programa actual del μ C mitjançant el botó "Read".

Drivers

En el CD del projecte també s'inclouen els drivers dels diferents dispositius que incorpora el robot. Aquests drivers faciliten la tasca de programació del robot, permetent que l'usuari comenci amb programes més senzills. Els drivers creats són:

- drvServo, pel control dels servomotors.
- drvIR, crea el senyal pel funcionament dels infrarojos.
- drvCompàs, per la utilització del compàs magnètic
- drvADC, driver del Conversor Analògic/Digital per fer funcionar els LDR.
- drvI2c, sistema de comunicació per la utilització de dispositius I2c, com el compàs magnètic.

Per utilitzar aquests drivers, només hem d'incorporar els arxius .c i les llibreries .h en el nostre projecte i utilitzar les funcions creades per la utilització dels dispositius.

RESUM :

En aquest projecte s'ha estudiat el disseny d'una plataforma robòtica mòbil per un PBL (Aprenentatge Basat en Problemes) en enginyeria informàtica. El principal objectiu és introduir aquest model en l'ensenyament universitari, com a complement de diferents assignatures de primer curso. Per arribar a aconseguir aquests objectius, s'ha dissenyat i construït una plataforma robòtica, dirigida per un microcontrolador i dotada de diversos sensors per interactuar amb l'entorn. El robot permet diferents tipus de programació i està especialment dissenyada per ser una bona experiència educativa.

RESUMEN :

En este proyecto se ha estudiado el diseño de una plataforma robótica móvil para un PBL (Aprendizaje Basado en Problemas) en ingeniería informática. El principal objetivo es introducir este modelo en la enseñanza universitaria, como complemento de diferentes asignaturas de primer curso. Para alcanzar estos objetivos, se ha diseñado y construido una plataforma robótica, dirigida por un microcontrolador i dotada de diversos sensores para interactuar con el entorno. El robot permite diferentes tipos de programación y está especialmente diseñada para ser una buena experiencia educativa.

ABSTRACT :

In this project, the design of a mobile robotic platform has been studied for a PBL (Project Based Learning) in Computer Engineering. The main objective is to introduce this model in university education, as a complement of different subjects in the first course. To achieve this goal a robotic platform was designed and built, it was directed by a microcontroller and it has different sensors to interact with the environment. The robot allows different types of programming and it is especially designed to be a great educative experience.